

Die Form der Datenbank.

A: Genealogien, Operationalitäten und Praxeologien  
relationaler Datenbanken in Ost und West.

B: Data Proxy – Documentation of an Artistic Data  
Dramatization Project

Ph.D. Thesis von Dipl.-Medienkünstler Francis Hunger

Vorgelegt am: 21.11.21

Disputation am: 25.11.22

Zur Erlangung des akademischen Grades Doctor of Philosophy (Ph.D.)

Fakultät Kunst und Gestaltung, Bauhaus Universität Weimar

Mentor/innen und Gutachter/innen

Prof. Dr. Gabriele Schabacher (Johannes Gutenberg-Universität Mainz)

Prof. Dr. phil. habil. Henning Schmidgen (Bauhaus Universität Weimar)

Prof. Yvonne Wilhelm (Zürcher Hochschule der Künste)

Prof. Christian Hübler (Zürcher Hochschule der Künste)



# Die Form der Datenbank. Genealogien, Operationalitäten und Praxeologien relationaler Datenbanken in Ost und West.

<b>1. Ausbreitung eines Forschungsfeldes</b>	<b>1</b>
1.1 Thesen	1
1.2 Geschichtete Felder in den Software-Studies	9
1.2.1 Kulturtechnikforschung	12
1.2.2 Infrastructure Studies	15
1.2.3 Kooperative Figuration	18
1.3 Historiographie im Spannungsfeld des Ost-West-Diskurses	24
1.3.1 Überlappende Zugänge	24
1.3.2 Periodisierungen	25
1.3.3 Aus dem Material heraus...	26
1.4 Archäologie des Immateriellen: Materialien und Methoden	29
1.4.1 Zeitungsartikel & Bedienungsanleitungen	29
1.4.2 Interviews	30
1.4.3 Archive	31
1.5 Navigation	32
<b>2. Genealogien relationaler Datenbanken</b>	<b>35</b>
2.1 Formatisierung – Strukturierte Daten in der relationalen Algebra	35
2.1.1 Die Nutzung ins Zentrum rücken	37
2.1.1.1 Information Retrieval	37
2.1.1.2 Nicht-Sequentieller Speicherzugriff	38
2.1.1.3 Abfragesprache mit Prädikatenlogik	41
2.1.1.4 Formalisierte Dateneingabe	42
2.1.1.5 Nutzungszentrierung	43
2.1.1.6 Kontext: Hyperlink oder Mengentheorie	43
2.1.1.7 Zusammenfassung	44
2.1.2. Namen oder Bedeutung	45
2.1.2.1 Entitäten im Spiegel der Universalismusdebatte	45
2.1.2.2 Zusammenfassung	49
2.1.3 Mengentheorie und Maschinen-Unabhängigkeit	50
2.1.3.1 Kontext: Timesharing und Virtuelle Adressierung	50
2.1.3.2 Maschinenunabhängigkeit und Relationale Mengentheorie	53
2.1.3.3 Zusammenfassung	55
2.1.4 Datenunabhängigkeit	56
2.1.4.1 Datenunabhängigkeit	57
2.1.4.2 Zusammenfassung	59
2.1.5 Die relationale Sicht auf Informationen und normalisierte Beziehungen	60
2.1.5.1 Menge	60
2.1.5.2 Modell, Schema, Relation	61
2.1.5.3 Operationen	67
2.1.5.4 Zusammenfassung	72
2.1.6 Fazit	72

2.2 Operationalisierung – Tabellieren	76
2.2.1. Tabellarische Materialität und Operationalität	76
2.2.1.1 Tisch – Tafel – Tabelle	79
2.2.1.2 Konstruktion und Information	81
2.2.1.3 Diagrammatiken – Zeitordnung und Raumordnung	88
2.2.1.4 Wissensoperationen – Den tabellaren Raum durchschreiten	90
2.2.2. Tabellarische Praktiken	93
2.2.2.1 Wissensordnungen – Die Literatur des Mittelalters in nationalen Kategorien	93
2.2.2.2 Statistik – Staatstabellen und Statistiken der Seidenproduktion	96
2.2.2.3 Transaktionen – Plantafeln: Prozess- und Organisationstabellen	101
2.2.2.4 Mathematische Tabellen – Menschliche Rechner, Logarithmen und die Massenproduktion mathematischer Ergebnisse	106
2.2.2.5 Zwischenfazit	117
2.2.3. Am Übergang zu Datenbankpraktiken	118
2.2.3.1 Prozessualität und Leere	119
2.2.3.2 Die Tabelle als Diskurs	122
2.2.3.3 Zwischen Abfrage und Menge: Oberflächen und Tiefenschichten	123
2.2.4 Fazit	125
2.3. Koordination – Transaktionen	128
2.3.1 Der Medial-Ökonomische Verbund der doppelten Buchhaltung	129
2.3.2 Die Depersonalisierung des Kapitals und dessen Verzeichnung	133
2.3.3 Verschränkungen kaufmännischer und statistischer Wissensfelder	137
2.3.4 Dispositive der Transaktionsverarbeitung	139
2.3.5 Ein Ticketverkäufer wird zum Informationsanbieter	143
2.3.6 On-Line Transaction Processing (OLTP)	147
2.3.6.1 Nicht-Linearer Speicherzugriff	147
2.3.6.2 Priorisieren, koordinieren und parallel verarbeiten	148
2.3.6.3 Von Stücklisten zu Kontrollsphären	152
2.3.6.4 System R – Ein relationales Forschungsprojekt	156
2.3.6.5 Technologietransfere und ›Benchmark Wars‹	159
2.3.6.6 Standardisierungen durch ACID	162
2.3.7 On-Line Analytical Processing (OLAP)	164
2.3.8 Fazit	168
<b>3. Technoemergenz im Ost/West-Vergleich</b>	<b>172</b>
3.1 Kontexte der Fallstudien	172
3.1.1 Institutionen der Planung und Lenkung in der DDR	174
3.1.2 Das Kombinat Robotron	178
3.1.3 Embargo des Coordinating Committee on Multilateral Export Controls	181
3.1.4 Software Adaptionen	183
3.1.5 Übersicht über Datenbanksysteme in der DDR	185
3.2 Robotron und SAP: Sozialistisches und kapitalistisches Enterprise Resource Management im Vergleich	189
3.2.1 Dresden & Walldorf: Den Produktionsprozess koordinieren	192
3.2.1.1 Explosionen: Bankspeicherung Technischer Informationen (BASTEI) / Bill Of Materials Processor (BOMP)	194
3.2.1.2 Automatisierte Maintenance mit COPICS und MAPICS	196
3.2.2 Dresden, Berlin(Ost): Digitalisierung und Vereinheitlichung	201
3.2.2.1 Pflichtenheft und Datenerhebung	201
3.2.2.2 Einheitliche Primärdokumentation	204
3.2.2.3 Integration in das Einheitliche System Elektronischer Rechentechnik (ESER)	207

3.2.3 Dresden, Guben, Moskau, Rostock: Sachgebietsorientierte Programmiersysteme (SOPS)	209
3.2.3.1 Von der Primärdokumentation zur modularen Softwarefiguration	209
3.2.3.2 Konzeption der SOPS	212
3.2.3.3 Einführung des SOPS BASTEI in drei Stationen	219
3.2.4 Walldorf: SAP und die modulare Integration	224
3.2.5 Dresden, Walldorf: Konzepte für die Formalisierung, Formatisierung und Informatisierung der Realität	234
3.2.6 Dresden, Berlin(Ost): Von der fließbandorientierten zur transaktionsorientierten Rationalisierung	241
3.2.7 Fazit	253
3.3 Kalifornien, Technische Universität Dresden und Robotron Dresden: Das Problemorientierte Seminar »Datenbanken« und DABA 1600	258
3.3.1 Ein Informatik-Zentrum der TU Dresden	260
3.3.2 Forschung und Entwicklung – die Zusammenarbeit von Robotron und TU Dresden	261
3.3.3 Ein Problemorientiertes Seminar als Knotenpunkt des Wissenstransfers	264
3.3.4 Forschungsergebnisse aus der Sektion Informationsverarbeitung	265
3.3.4.1 Vorarbeiten	266
3.3.4.2 Dissertationen	272
3.3.4.3 Realisierung	277
3.3.5 Übergang von der Forschung und Entwicklung in die Produktion	280
3.3.6 Anwenderinformationen	281
3.3.7 Fazit	288
<b>4. Kooperative Medienpraktiken der Datenbank</b>	<b>293</b>
4.1 Ergebnisse des untersuchten Materials	293
4.2 Konsequenzen für die Geschichte des Computings und die Software Studies	302
4.3 Anschlussfragen	306
4.4 Coda: Deine Rücksendung wurde abgeholt.	311
<b>Bibliographie</b>	<b>313</b>
<b>Abbildungsverzeichnis</b>	<b>342</b>
<b>Abkürzungsverzeichnis</b>	<b>345</b>
<b>Originalsprachliche Zitate</b>	<b>347</b>



# 1. Ausbreitung eines Forschungsfeldes

»Um 1964 trat in der Computerliteratur ein neuer Begriff auf, um ein neues Konzept zu markieren. Der Begriff war ›Daten Bank‹, und er wurde von Leuten gefasst, die an militärischen Informationssystemen arbeiteten, um Datensammlungen zu bezeichnen, welche durch die Endnutzer von Time-Sharing Computersystemen geteilt wurden« (McGee 1981, 505).<sup>1</sup>

## 1.1 Thesen

Wo von Software die Rede ist, sind relationale Datenbanken immer bereits vorausgesetzt. Relationale Datenbanken ordnen die Welt und ermöglichen Kooperation. Datenbankanwendungen kommen dort zum Einsatz, wo es gilt, Daten zu verarbeiten – kurz gesagt: überall. Was ist eine relationale Datenbank? Was sind ihre Entstehungsbedingungen als Medienobjekt, als Diskurs und als Praxis? Welche historischen Genealogien lassen sich für ihre Entstehung nachzeichnen und welche Folgen hat dies für die Einordnung von Datenbankpraktiken als Medienkulturtechnik?

Diese Untersuchung entfaltet sich aus drei Perspektiven. Im Teil I werden die Genealogien relationaler Datenbanken anhand dreier Felder untersucht: der Formatierung von Daten, der Operationalisierung von Daten am Beispiel von Tabellenpraktiken und der Datenpraxis der Transaktion. Im Teil II wird eine Softwaregeschichte der (relationalen) Datenbank anhand von Betriebsverwaltungssoftware der 1960er bis späten 1980er Jahre entworfen, und zwar in einem Ost-West-Vergleich aus ›sozialistischen‹ und ›kapitalistischen‹ Datenbanken. Der Teil III ist ein Kunstprojekt, welches Datenpraktiken nicht nur untersucht, sondern selbst auch hervorbringt. Auf einer Website und durch eine Reihe von YouTube-Vlogs wird die alternative Realität des ›Dataproxy‹ konstruiert, und der künstlerische Prozess in einer gesonderten schriftlichen Dokumentation dargelegt.

Erste Überlegungen verorten Datenbanken im Feld der Archive.<sup>2</sup> Viele halten sie für Werkzeuge der Überwachung<sup>3</sup> oder setzen sie in Eins mit Suchmaschinen oder Web 2.0-Anwendungen. In soziologischen Texten werden sie häufig schlichtweg unter ›Algorithmus‹ subsumiert.<sup>4</sup> Ein Großteil der Programmier-Literatur und der Informatik-Ausbildung stellt Datenbanken als rein technische Artefakte

---

<sup>1</sup> Für eine bessere Lesbarkeit des Textes wurden alle fremdsprachigen Zitate in die Textsprache übertragen. Die originalsprachlichen Zitate sind, durch lateinische Ziffern markiert, als Endnoten dem Text angefügt. Im Text wird möglichst geschlechter-sensibel formuliert. Aufgrund der ohnehin hohen Dichte und Komplexität des Textes wurde nach verschiedenen Versuchen auf das Gendern durch \* verzichtet.

<sup>2</sup> Zu Archiven siehe *Archivprozesse – die Kommunikation der Aufbewahrung* (Pompe/Scholz (Hrsg.) 2002), *Archivologie – Theorien des Archivs in Wissenschaft, Medien und Künsten* (Ebeling/Günzel (Hrsg.) 2009) und *AnArchive(s) – eine minimale Enzyklopädie zur Archäologie und Variantologie der Künste und Medien* (Giannetti/Zielinski (Hrsg.) 2014).

<sup>3</sup> Bedeutendste Vertreterin dieser Auffassung ist Zuboff unter dem Schlagwort des Surveillance Capitalism (Zuboff 2015). Ihr hat Philip E. Agre 1994 mit ›Capture‹ (Erfassung) eine anderes Konzept entgegen gesetzt, welches durch Till Heilmann weiter entwickelt wurde (Agre 1994; Heilmann 2015). Zur Kritik an Zuboffs einseitiger Konzentration auf Surveillance *Capitalism's New Clothes* (Morozov 2019a).

<sup>4</sup> Das Problem ›Datenbankpopulismus‹ formulierte Driscoll 2012, als er anmerkte, dass Nicht-Spezialisten dazu neigen, die Ausprägung großer Datensammlungen entweder zu über- oder unterschätzen (Driscoll 2012, 1).

der Softwareproduktion dar.<sup>5</sup> Auffällig ist: Immer schon scheint die Datenbank gegeben und existent, so wie der Igel zum Hasen ruft: »Bin schon all hier«. Entsprechend bleiben Datenbanken unsichtbar, tief eingegraben in Schichtungen von Programmiersprachen und Softwareoberflächen oder (fälschlicherweise) als vernachlässigbar subsumiert unter dem Oberbegriff ›Algorithmus«. Findet sie dezidierte Beachtung, vorrangig in technischer Literatur, erscheint die Datenbank als ein rein technisch bestimmtes, leidenschaftsloses, dienstfreudiges und politisch neutrales Artefakt, welches dem Gebiet der Computerwissenschaften zuzuschlagen sei.

Archiv, Sammlung, oder Suchmaschine – Konzepte, welche die Datenbank beständig umkreisen, werden hier nicht als Ausgangspunkte gesetzt. Stattdessen soll eine spezifische Softwarefiguration untersucht werden, die in der technischen Fachliteratur als ›Datenbanksystem‹ bezeichnet wird. Das betrifft zwei wesentliche Aspekte: Zum einen umfasst das Datenbanksystem die Datenbanksoftware, häufig als Datenbankmanagementsystem (DBMS) bezeichnet, welche eine grundsätzliche Operationalität für den Umgang mit Daten zu Verfügung stellt. Zum anderen betrifft es ›die Datenbank‹, im Sinne von Datenbankanwendungen, welche konkrete Aufgaben für eine Organisationseinheit, eine bestimmte Branche oder eine Domäne bewältigen soll. Diese flexiblen Figuren dienen verschiedensten Verwendungszwecken, zum Beispiel als betriebliche Software für Produktions- und Dienstleistungen, im Bibliothekswesen und für wissenschaftliche Anwendungen oder im Bereich der Verwaltung und Organisation und der Logistik.

In diesem Zuge ist zu konstatieren, dass Datenbanken intensiv in kaskadierende Softwarefigurationen verflochten sind. Daraus ergibt sich eine größere Behauptung: Wo von Software die Rede ist, sind Datenbanken immer schon mitgemeint. Weil Datenbanken grundlegende Funktionen für andere Softwareschichten zur Verfügung stellen und darin eingebettet sind, handelt es sich bei Datenbanken um Infrastruktur. Es folgt die Frage, wodurch Datenbanken Infrastruktur werden. Gibt es spezifische infrastrukturelle Momente der Datenbank, die zu ihrer durchdringenden Verbreitung beitragen?

Die äußerst mannigfaltigen Verwendungszwecke verdeutlichen, dass relationale Datenbankmanagementsysteme in sehr verschiedene Anwendungsbereiche überführt werden können. Woher kommt diese Fähigkeit zur Pluralität? Über welche Potentiale der Einbindung und Anpassbarkeit verfügen Datenbankbetriebssysteme? Wie formieren und formatieren Menschen diese komplexen technologischen Gebilde? Datenbanken sind nicht Produkt, sondern Prozess: Ihr Gewordensein ist in den Blick zu nehmen, denn für die jeweiligen Anwendungsbereiche wurden sie angepasst, erweitert und verknüpft, so dass sie Teil einer größeren Figuration werden – einer Datenfiguration.

Der aktuelle Hype um Künstliche Intelligenz ignoriert, dass diese für sich genommen keinen wesentlichen Faktor und keine neue Stufe der Automatisierung darstellt. Vielmehr ist sie in lang existierende technologische Automatisierungen und Formalisierungen eingebettet (vgl. Moore 2019). Künstliche ›Intelligenz‹ betrifft jene Bereiche, welche durch bisherige Automatisierungen nur zum Teil maschinisiert werden konnten, vor allem im Bereich der Mustererkennung. Doch die zugrunde liegenden Infrastrukturen existieren bereits lange. Es fehlte angesichts dessen an einer durchdringenden, medientheoretischen Würdigung der Gray Media und Logistical Media, welche die Wirkung von Software-Mensch-Figurationen in der Automatisierung hinreichend erklären würde.

---

<sup>5</sup> Stellvertretend für das Genre der Programmierhandbücher seien genannt *Joe Celko's data & databases – concepts in practice* (Celko 1999), *Database Management Systems – A Business-Oriented Approach using ORACLE, MySQL and MS Access* (Zygiaris 2018) and *Beginning PHP and MySQL* (Kromann 2018).



Ähnlich wurde die Emergenz des Internet und des WWW im populären Diskurs als »Information Super-Highway« (Gore 1994) euphorisch markiert, und im akademischen Feld als »Ausgangspunkt einer grundlegenden kulturellen Transformation« (Castells 2010, xviii),<sup>ii</sup> und als Moment des »Netzwerkka-pitalismus« (Fuchs 2008, 104)<sup>iii</sup> kritisch thematisiert. Eine vergleichbar intensive Rezeption von Datenbanken und Datenbankmanagementsystemen findet bisher nicht statt. Das konzeptuelle Verständnis von »Datenbank« blieb allgemein. Der Begriff wird häufig für allgemeine Datensammlungen jeglicher Art verwendet. In seiner grundlegenden kulturellen, technologischen und gesellschaftlichen Bedeutung als Medium der Koordination ist das Konzept »Datenbank« bisher unzureichend erforscht und dargestellt. Die historisch informierten Einsätze zur Emergenz der Datenbank durch Thomas Haigh (Haigh 2001; Bachman 2004; Haigh 2006; Bergin/Haigh 2009; Haigh 2009; Haigh 2012), als auch die tiefgehende medientheoretische Einordnung durch Markus Burkhardt (Burkhardt 2015) haben geholfen, diesen Umstand zu lindern. Sie hinterließen allerdings die Aufgabe, elektronische Datenbanken aus der Perspektive der Software Studies und der Kulturtechnikforschung als Infrastruktur und Kulturtechnik der Kooperation zu erkunden. Im Anschluss an Wendy Chun, die fragte: »Why networks, why now?« (Chun 2014 at 9:32 min), stellt sich die Frage: Warum Datenbanken, warum jetzt?

Mit dieser Justierung wird die Datenbank als soziotechnisches Gemenge befragt, denn so Conrad »Medien und Technologien sind vermengt mit Organisationsprozessen und dem Wissen und den Fähigkeiten der Nutzer« (Conrad 2017, 189). Es wird zur Diskussion gestellt, welche Einsätze in ihre Konzeption, Umsetzung, Ausführung und Wartung fließen. Für das »Wie« der Datenbank gilt es zu untersuchen, wie die medialen Qualitäten verflochten sind in die Praktiken<sup>6</sup> der Verwendung. Warum wird eine große Breite verschiedener Praktiken in dieser Figuration ermöglicht? Dem ist auf zweierlei Art nachzugehen: Zum einen, indem historische Genesen relationaler Datenbanksoftware und Datenbankanwendungen nachgezeichnet werden. Zum anderen werden Fallbeispiele befragt, in deren Rahmen Praktiken des Umgangs mit Daten in Datenbanken medial und kulturtechnisch zu erörtern sind. In diesen Fällen kann das Wechselspiel aus menschlicher und automatisierter Handlungsmacht befragt werden, denn nur im Zuge dieses Wechselspiels erzeugen Datenbankfigurationen überhaupt neues Wissen.

Das Gewordensein der Datenbank im Spannungsfeld humaner und non-humaner Agency wirft auch die Frage zur Informationsrealität der Datenbank im Verhältnis zur Alltagsrealität auf. Menschen bestimmen die Informationsrealität der Datenbank, indem sie Entscheidungen über den Informationsgehalt treffen, die in der Datenbankstruktur eingeschrieben werden. Die Frage nach dem Gewordensein der Datenbank schließt somit die Frage nach den Entscheidungen, die zur Festschreibung des Informationsmodells führen, ein.

Auch wenn es auf den ersten Blick so erscheinen mag: Diese Entscheidungen sind nicht frei. Sie sind vorfiguriert durch die Datenlogik, welche vom Datenbankbetriebssystem zur Verfügung gestellt wird. Heute werden, neben den historisch bedeutsamen Datenlogiken hierarchische Logik<sup>7</sup> und Netzwerk-

---

<sup>6</sup> Johanna Seifert hat in *Praktiken, Techniken, Operationen – Die praxeologische Wende in der Medien- und Kulturwissenschaft* (2019) ein Praxiskonzept dargestellt, welches nicht mehr allein auf rein menschliche Subjektivität und Intentionalität rekurriert. Vielmehr würden inzwischen »kollektive und relationale Moment menschlichen Handelns« betont, indem »sowohl auf die Materialität sozialer Beziehungen [...], als auch auf die Entstehungsprozesse« aufmerksam gemacht werde (Seifert 2019, 102).

<sup>7</sup> Historisch bedeutsame Vertreterin der hierarchischen Datenlogik ist das IBM Information Management System (IMS) seit Mitte der 1960er Jahre. Die Daten waren als voneinander hierarchisch abhängige Knoten nach dem Eltern-Kind-Prinzip organisiert.

Logik,<sup>8</sup> sowie aktuellen Entwicklungen, wie Schema-freien Logiken<sup>9</sup> und Big-Data-Konzeptionen,<sup>10</sup> drei Viertel der derzeit eingesetzten Datenbankmanagementsysteme durch das relationale Modell strukturiert (DB-Engines/Solid IT 2020).<sup>11</sup> Die relationale Algebra soll, ausgehend von ihrer heutigen hohen Verbreitung, den Fokus dieser Untersuchung bilden.

Eine Figur, der wir in der Untersuchung häufig begegnen werden, ist die Frage nach der Form der Datenbank. Um den Blick auf die Form und Formation zu lenken, wird dieser Umstand gesondert sprachlich gekennzeichnet und es ist von ›Information als Formation‹ die Rede.

Information, Daten, Datenlogiken und Informationsrealität sind damit als Formationen markiert. Die weit reichenden Konsequenzen werden im Laufe der folgenden Kapitel herausgearbeitet. Anzudeuten ist bereits, dass sie die Dominanz des Shannon/Weaver Informationsmodelles für die Medientheorie in Frage stellen.

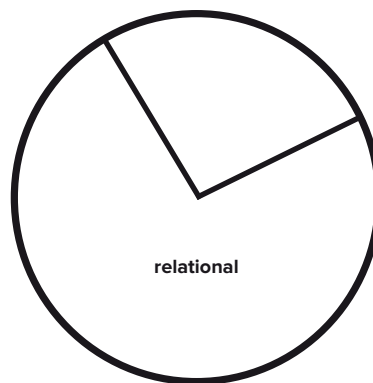


Abb. 1: Die Mehrzahl der weltweit eingesetzten Datenbankmanagementsysteme sind derzeit relational organisiert. Der Anteil der nicht-relationalen DBMS konstituiert sich aus Graph DBMS, Document Stores, Key-Value Stores, Suchmaschinen und Wide Column Stores. Den größten Zuwachs der letzten Jahre verzeichnen Graph DBMS, die jedoch insgesamt gesehen derzeit bei 1,5% Anteil liegen (Autor, nach DB-Engines/Solid IT 2020).

Zur Fragestellung trug auch die berufliche Praxis des Autors als Programmierer bei. Die Frage danach, was eine Datenbank überhaupt ist, stellte sich überhaupt erst im Zuge der Umsetzung von Websites, Content Management Systemen, Enterprise Resource Planning und Customer Relationship Management Anwendungen. Denn, so beredt die verfügbaren Dokumentationen sich zu den Sprachelementen der Structured Query Language (SQL), Fragen der Datenorganisation, oder zur Installation von DBMS äußern, so sehr beschweigen sie, woher die Konzepte stammen, warum sie entwickelt wurden und auf welche Weise sie die nunmehr verfügbare Form angenommen haben.<sup>12</sup> Der sich in diesem Schweigen ausdrückenden Maxime »Bin schon all hier« ist die Fragestellung entgegen zu stellen: »Wie wurde die relationale Datenbank?«.

---

<sup>8</sup> Der Integrated Data Store, entwickelt in den frühen 1960er Jahren in den USA gilt als erste Datenbank-Figuration nach dem Netzwerkmodell. Sein Entwickler Charles Bachmann prägte den generellen Datenbank-Diskurs der 1960er Jahre.

<sup>9</sup> Derzeitige Vertreterinnen sind MongoDB, Amazon DynamoDB, Couchbase, Microsoft Azure Cosmos DB, CouchDB.

<sup>10</sup> Hier sei stellvertretend Apache Hadoop unter Einbeziehung des Mapreduce-Algorithmus genannt.

<sup>11</sup> Zu den meistgenutzten relationalen Datenbanken zählen die Produkte Oracle, MySQL, Microsoft SQL Server, PostgreSQL, IBM DB2, MariaDB, Filemaker und SAP Hana. Daneben gibt es hunderte spezialisierte relationale Datenbankmanagementsysteme, die im Einsatz sind.

<sup>12</sup> Generell zeichnen sich ein Großteil der Programmierliteratur, zu nennen seien beispielsweise Programmiersprachen und Betriebssysteme als weitere große thematische Blöcke, durch eine Abwesenheit von Informationen zur Genese oder zur gesellschaftlichen Situierung aus.

Historisch ist die relationale Datenbankfiguration im Übergang vom Großcomputer zum Personal Computer und im Zuge zunehmender Miniaturisierung der Computer und Durchdringung des Alltags mit informationsverarbeitenden Maschinen zu verankern.<sup>13</sup> Erste relationale Konzepte entstanden in den frühen 1970er Jahren. Bis zur breiten Durchsetzung dauerte es bis Anfang der 1990er Jahre. Ihre zunehmende Präsenz fällt in einen Zeitraum, der als Übergang der Disziplinargesellschaften zu Kontrollgesellschaften beschrieben wurde, einhergehend mit der Flexibilisierung gesellschaftlicher Rahmenbedingungen,<sup>14</sup> mit zunehmender Globalisierung und darin verwickelten Medientechnologien<sup>15</sup> und schließlich auch mit Zweifeln an der epistemologischen Funktion der Kritik.<sup>16</sup>

Die 1990er Jahre sind geprägt durch das Ende des Kalten Krieges, welcher als treibend für zahlreiche technologische Entwicklungen (prominent: Raumfahrt, Atombombe, Personalcomputer, Internet) seit Ende des Zweiten Weltkrieges beschrieben wurde. Gerade der Erfolg des Internets wurde in einen Zusammenhang mit der zunehmenden Globalisierung gestellt und medienhistorisch eingereiht in die Verbreitung der Kommunikationsnetzwerke Eisenbahn, Morsecode und Telefonie. Somit bildeten Netzwerke einen Fokus der medientheoretischen Forschung, die den Übergang zur Erforschung von Software bildete.<sup>17</sup> Dass an den Enden der Kommunikationsnetzwerke die eingehenden Informationen prozessiert werden mussten, ist zwar thematisiert worden, allerdings wird im öffentlichen Diskurs die immense Bedeutung von Datenbanken für transaktionale Informationen und als Medien der Kooperation weitestgehend ignoriert.

Den Transaktionen, also dokumentierten und umkehrbaren Übertragungen, kommt im Zusammenhang mit Datenbankpraktiken eine nicht zu unterschätzende Bedeutung zu. Ohne die Transaktionsverarbeitung in Datenbanken wären die ›Information-Highways‹ kaum operationalisierbar gewesen: Internationale Geldtransfers, globaler Flugverkehr und weltumgreifende Warenlogistik sind neben den sichtbareren Kommunikationsplattformen des Web 2.0 eindrückliche Zeugen der transaktionalen Globalisierung mit und durch Datenbanken. In dieser Lesweise war das relationale Datenbankmodell medial beteiligt an der Rekonfiguration kapitalistischer Ökonomie in Folge der Industrialisierungskrisen der 1970er in Europa und den USA, einschließlich dem Ende des industriellen Sozialismus der 1990er Jahre in Osteuropa und dem gegenwärtig andauernden industriellen Aufstieg asiatischer Staaten.

Yuk Hui hinterlässt in seiner Monographie zu digitalen Objekten ein paradigmatisches Beispiel dafür, wie der zeitgenössischen Medienphilosophie und Medientheorie die Bedeutung von

---

<sup>13</sup> Siehe *A history of modern computing* (Ceruzzi 2003) und *A veritable bucket of facts – origins of the data base management system* (Haigh 2006). Dabei ist unbedingt zu vermerken, dass Übergang nicht die Ablösung des einen durch das andere meint. Vielmehr wurden Großcomputer, Kleincomputer und Personal Computer parallel verwendet und die damit einhergehenden verschiedenen Nutzungsweisen griffen und greifen ineinander (vgl. Haigh 2018, 2).

<sup>14</sup> Siehe *The postmodern Condition – A Report on Knowledge* (Lyotard/Jameson 1984), *We have never been modern* (Latour 1991) und *Liquid modernity* (Bauman 2000).

<sup>15</sup> Siehe *The rise of the network society* (Castells 2010). Zur Lean Production und Just in Time siehe *Immaterielle Arbeit – Ästhetisierung der Politik und der Produktion unter den Bedingungen des Postfordismus* (Lazzarato 1998) und *Software, infrastructure, labor – a media theory of logistical nightmares* (Rossiter 2016).

<sup>16</sup> Siehe *We have never been modern* (Latour 1991), *A new Philosophy of Society – Assemblage Theory and Social Complexity* (De Landa 2006) und *Eine neue Soziologie für eine neue Gesellschaft: Einführung in die Akteur-Netzwerk-Theorie* (Latour 2010).

<sup>17</sup> Siehe *Protocol – how control exists after decentralization* (Galloway 2004), *Toward Information Infrastructure Studies – Ways of knowing in a Networked Environment* (Bowker u. a. 2010) und *How not to network a nation – The uneasy History of the Soviet Internet* (Peters 2017).

Transaktionsdatenverarbeitung entgeht. Er führt das Konzept der digitalen Milieu wie folgt ein: »We are currently living in a digital milieu; we Facebook, we blog, we Flickr, we YouTube, and we Vimeo. Nouns and brands have become verbs, even forms of life« (Hui 2016, 47). Übersehen wird in dieser Aufzählung des digitalen Milieus eine ganze Reihe von Dienstleistungen, Services, Produktionsweisen und Logistiken, die, auch wenn sie nicht verbisiert wurden, mindestens genauso grundlegend sind, in anderen Worten: We SAP, we Oracle, we Salesforce, we Databank.

Eine sichtende Auseinandersetzung mit der vorhandenen Historisierung von Datenbanken – hauptsächlich betrieben durch den Historiker Thomas Haigh – erfährt das Entstehen dieses Wissenswerkzeugs als USA-zentriert. Lässt sich diese Erzählung dezentrieren, lässt sie sich internationalisieren? Der vergleichende Blick hinter den Eisernen Vorhang, in das Geburtsland des Autors, die DDR, soll Klärung bringen: Wenn es ›kapitalistische‹ Datenbanken gibt, gab es dann ›sozialistische‹? Wo ähnelten sich die Nutzungen, wo unterschieden sie sich? Wie war das Verhältnis von Technologietransfer und Eigenentwicklung in der DDR und was sagt dies über das Konzept ›relationale Algebra‹ aus? Unterschied oder glich sich die Art und Weise der Operationalisierung? Welche Schlüsse folgen daraus? Welche Konzepte der relationalen Datenbank sind in anderen medialen Ausformungen zu finden und in die Datenbank übersetzt worden? In der Beantwortung dieser Fragen kann das vorliegende Vorhaben einen Beitrag zum Feld der Software-Studies leisten, welcher die West-Zentrierung bisheriger Forschung ergänzt und auf deren blinde Flecken aufmerksam macht. Darauf, dass dieses Desiderat einer nicht auf den Westen zentrierten Historisierung nicht nur ein medienhistorisches, sondern auch ein medientheoretisches ist, weist Schüttpelz dezidiert hin (Schüttpelz 2016, 14). Dieser Befund soll an zwei Beispielen verankert werden: Paul Ceruzzis Einführung *A History of Modern Computing* (2003) ist in weiten Teilen US-zentriert. Die Dissertation *Geschichte der Softwarebranche in Deutschland* (Leimbach 2009) genügt sich auf 461 Seiten ohne die geringste Erwähnung der DDR-Informatik oder DDR-Softwareproduktion, obgleich sie »Deutschland« im Titel trägt.

Für eine Mediengenealogie der Datenbank leistet diese Studie einen grundlegenden Beitrag. Im Vergleich der beiden gesellschaftlichen Konfigurationen Ost und West kann anhand von Fallbeispielen eine überaus deutliche Zentralstellung der Datenbankpraktiken als mediale Praxen der Koordination belegt werden. Damit wird das Konzept ›Datenbank‹ repositioniert, jenseits eines einseitigen Verständnis als Kommunikationsmedium. Die medialen Praxen der Koordination standen in Ost und West unter verschiedenen ideologischen Vorzeichen: auf der einen Seite als Rationalisierung zu verrichtender Arbeit zum sozialistischen Systemerhalt, auf der anderen Seite als Optimierung von Informationsflüssen im Sinne kapitalistischer Konkurrenz. Doch zeigen sich auf medienhistorischer Ebene deutliche Parallelen der Datenbankfigurationen: technologisch, infrastrukturell und praxeologisch. Der gemeinsame Kontext der Industrieproduktion, so ließ sich anhand der Enterprise Resource Planning-Software zeigen, war deutlich prägender, als unterschiedliche ideologische und ökonomischen Rahmungen.<sup>18</sup>

Bisher fehlten für diese These jene Belege, die nun im Rahmen dieser Arbeit erbracht werden.

Solche Überlegungen rufen die Notwendigkeit auf, die vorhandene Literatur in Bezug auf einen Ost-West-Fokus kurz zu skizzieren: Die westliche Entwicklung und Bedeutung relationaler Datenbankmanagementsysteme ist in einer Reihe von Aufsätzen historisch aufgearbeitet worden. In den IEEE Annals

---

<sup>18</sup> Historisch lässt sich diese Überschneidung im Feld der Industriegesellschaft belegen. Schließlich war Lenin, der Russland ab 1917 vom Agrarstaat zum sozialistischen Industriestaat katapultieren wollte, ein Anhänger des wissenschaftlichen Managements nach Frederick W. Taylor (Scoville 2001), und begeistert von den preußischen Planungsleistungen im 1. Weltkrieg (Brie 2017, 40), welche die Idee einer fordistischen Planwirtschaft, nur eben sozialistischer Natur, wesentlich inspirierten (Castells 2010, 462).

of the History of Computing 2012, vol. 4, gaben Pioniere dieser Industrie Auskunft über die frühen Entwicklungen bei IBM, Oracle und der University of California, Berkeley (Wade/Chamberlin 2012; Darwen 2012; Grier 2012). Haigh lieferte zahlreiche Einzelbeiträge und führte Oral History Interviews mit Protagonisten von relationalen und nicht-relationalen Datenbankmanagementsystemen (Bachman 2004; Haigh 2006; Date 2017; Haigh 2009; Haigh 2012).

Die vorliegende Studie ist neben *Computing in Russia – the history of computer devices and information technology revealed* (Trogemann/Nitussov/Ernst (Hrsg.) 2001), *Setun – eine Recherche über den sowjetischen Ternärcomputer* (Hunger 2007), *Cybernetic Revolutionaries – Rechnology and Politics in Allende's Chile* (Medina 2014) und *How not to network a nation – The Uneasy History of the Soviet Internet* (Peters 2017) eine der wenigen medientheoretisch informierten Untersuchungen, welche die osteuropäische beziehungsweise realsozialistische ›Tiefenzeit der Medien‹ (Zielinski 2002) ernsthaft untersuchen.<sup>19</sup> Keiner der genannten Texte adressiert Datenbanken und deren Genese im sozialistischen Kontext, sodass hiermit ein Desiderat behoben werden kann.

Die Limitationen der Untersuchungen von Medina (2014) und Peters (2017) liegt darin, dass sie die Beschreibungsebene der Alltagspraxen mit Daten ignorieren und vorrangig einem doppelten Phantasma der Kybernetisierung nachgehen.

Das Phantasma der Kybernetisierung betrifft einerseits den Zeitraum der Historisierung von Informationsverarbeitungspraktiken. Kybernetik, ein Geflecht verschiedener Fachdisziplinen (Nachrichtentechnik, Spieltheorie, Systemtheorie, Anthropologie und Psychologie), entstanden ab den 1940er Jahren, wird unter dem Aspekt der als Feedback verstandenen Steuerung und Regelung im Zeitraum des Kalten Krieges (1947 bis 1989) positioniert. Im Unterschied dazu wird hier argumentiert, dass die Periode des Kalten Krieges als Beschreibungsebene nicht ausreicht und zeitlich und konzeptuell breiter zu fassen ist. Die Auseinandersetzung mit den Genealogien von Datenpraktiken und Datenbankpraktiken reicht mindestens bis in das frühkapitalistische 15. Jahrhundert zurück, welches sich durch eine zunehmende Bedeutung von Registrierungstechniken wie der Tabelle, der doppelten Buchhaltung oder der Bibliothek auszeichnet. Mit der Industrialisierung ab dem 18. Jahrhundert werden diese und weitere bürokratische Medienpraktiken Teil einer grundlegenden Infrastruktur, die bis heute unseren Alltag durchzieht. Eine an Datenpraktiken orientierte Praxeologie ist in der Lage, eine andere Tiefenzeit zu beschreiben, als eine kybernetische Rahmung. Dies führt außerdem zu theoretischen Konsequenzen, welche Fragen der Informationstheorie betreffen und weiter unten ausgeführt werden.

Das Phantasma der Kybernetisierung besteht außerdem darin, dass das sowjetische OGAS in weitaus geringerem Maße realisiert wurde als geplant, und das chilenische Cybersyn allenfalls als ein kurzzeitiges Experiment gelten kann. Beide wurden auf der Ebene persistenter medialer Praxen, die sich über Jahrzehnte oder Jahrhunderte strecken kaum wirksam. Jenseits des Registers des Imaginären sagen OGAS und Cybersyn aufgrund der Kürze ihres Bestehens weniger über nicht-kapitalistisches Computing aus, als erhofft. Dies wird heute deswegen relevant, weil die aktuelle Socialist Calculation Debate vor allem Cybersyn als Praxisbeispiel heranzieht (siehe zum Beispiel in: Morozov 2014; Morozov 2019b; Jochum/Schaupp 2019).

In der vorliegenden Arbeit werden im Unterschied dazu Datenpraktiken wie die Sachgebietsorientierten Programmiersysteme (SOPS), welche zwanzig Jahre lang in der DDR im Einsatz waren, betrachtet.

---

<sup>19</sup> In der künstlerischen Praxis des Verfassers spielen Untersuchungen von Wissenskonstellationen und eine Kritik der Historiographie im Ost-West-Verhältnis unter anderem in den Arbeiten *The Setun Conspiracy* (2005–2008), *History Exhaustion* (2009) und *Utopia Generator* (2015) eine wesentliche Rolle.

So geht die Studie einer übergeordneten Fragestellung nach, welche als mediengeschichtliche Frage formuliert wurde, nicht als ökonomische oder politische. Davon ausgehend, dass der Einsatz von Datenbanken durch die Verdichtung von Informationsflüssen eine zunehmende Flexibilisierung in Verwaltung, Management und Produktion ermöglicht, steht zur Debatte: War das Ende der DDR darin begründet, dass eine Umstellung von vorrangig industriellen, fordistischen Produktionsweisen auf eine zunehmend post-fordistische, von Dienstleistungen geprägte Ökonomie nicht gelang? Ließe sich die Bedeutung relationaler Datenbankmanagementsysteme markieren, indem ihre Abwesenheit, oder ihr später Einsatz (in der DDR ab 1985 und signifikant ab 1987) thematisiert wird? Diese Fragestellung mag erst einmal überraschen. Um es vorwegzunehmen: Zu viele verschiedene Faktoren spielten beim Aufstieg und Fall des Staatssozialismus eine Rolle, und die Frage bleibt in dieser Form unbeantwortbar. Doch diese Zuspitzung ergibt einen Nachweis für die enge Verflochtenheit post-fordistischer Ökonomien und relationaler Datenbankpraktiken.

Die Frage nach der Materialität von Datenbanken führt, so scheint es zunächst, in unmittelbar technologische Gefilde. Wenn aber technologische Konzepte, unter Beteiligung von humanen und nicht-humanen Akteuren als soziale Praxen gelesen werden, dann schälen sich andere Fokoi heraus: Immer wieder kreuzen die Materialitäten der Tabelle 1.) als Visualisierung mathematischer Mengen, 2.) als Zuschreibung von Referenz und 3.) als Wissensversammlung spezifischer Domänen den Weg. Anhand der Art und Weise, wie in der Mengenalgebra und in Tabellen digitale Referenz (auf die Wirklichkeit) erzeugt wird, lässt sich aufzeigen, dass das Shannonsche Informationsmodell von Sender-Empfänger-Signal-Rauschen (Shannon 1948) weitestgehend an der Beschreibungsebene von Datenbanken vorbei zielt (vgl. auch Burkhardt 2015, 167f.).

Geht man von der grundlegenden Durchdringung aktueller Software durch (relationale) Datenbankmanagementsysteme aus, wird deutlich, dass ein Signal-Rauschen-orientierter Informationsbegriff nur unzureichend geeignet ist, aktuelle Datenpraktiken zu beschreiben. Es ist eine zutreffenderer Informationsbegriff empirisch zu grundieren und theoretisch heraus zu arbeiten, welche sich davon unterscheidet. Damit öffnet sich der Weg, Datenbanken und die mit ihnen verknüpften sozialen Aktionen und Operationen, als Medien der Kooperation zu fassen, welche durch ein spezifisches Informationsmodell gekennzeichnet sind.

Aus diesen Überlegungen schälen sich vier Hauptthesen heraus, welche die vorliegende Untersuchung orientieren und von der bereits vorhandenen Forschung unterscheiden:

1. Die Emergenz der relationalen Datenbank ist an der Schwelle der ökonomischen globalen Vernetzung des spätindustriellen Kapitalismus, sowie am Übergang von Disziplinar- zur Kontrollgesellschaft beobachtbar. Sie ermöglicht internationale Waren-, Menschen- sowie Informationsströme und ist grundlegendes Medium der Kooperation in einer globalen Infrastruktur der Informationsverarbeitung.
2. Relationale Datenbanken sind Teil eines kulturtechnisch verankerten emergenten Komplexes der Wissensorganisation. Neben ihrer mathematisch-logischen Operationalisierung basierend auf der Mengentheorie, fügen das Tabellieren und die prozessualen Praktiken der Transaktion der Symbolverarbeitung in und mit Software neue, kollaborative Register hinzu.
3. Relationale Datenbanken sind im Unterschied zu Künstlicher ›Intelligenz‹ oder zu Netzwerken genealogisch nicht vordergründig mit signalorientierten, kybernetischen Konzepten von Information verknüpft. Stattdessen entsammen sie registrierenden und koordinierenden

Alltagspraxen, woraus die Notwendigkeit eines adäquaten Konzeptes der ›Information‹ als formatiert. Dieses wird hier als ›In Formation‹ beziehungsweise ›Information als Formation‹ positioniert.

4. Der historisch vergleichende Blick auf die Emergenz relationaler Datenbanken im Westen und Osten am Übergang vom Mainframe zum Personal Computer kann mediale Spezifika in Konzeption und Gebrauch offenlegen und damit auch zu einer Schärfung des konzeptuellen Verständnisses des Computings beitragen.

Auf die Darstellung der Forschungsstände soll an dieser Stelle verzichtet werden, diese sind in den ersten Absätzen der jeweiligen Kapitel referenziert. Stattdessen wendet sich die Einleitung den Theorien und Methoden zu, welche die vorliegende Arbeit begleitet haben.

## 1.2 Geschichtete Felder in den Software-Studies

»Es gibt eine Schnelligkeit, Idiosynkrasie und Thickness in den Veränderungen, die durch Software bewirkt wurden, welche sie zu einem grundlegend komplizierten Phänomen machen, potentiell reichhaltig, anstatt inhärent reduktiv, jedoch nicht notwendigerweise« (Matthew Fuller, In: Chun 2011, viii).<sup>iv</sup>

Grundsätzlich sind der erste und zweite Teil hinsichtlich der verwendeten Ansätze zu unterscheiden: Der erste Teil verwendet Methoden der Software Studies, der Diskursanalyse, der Kulturtechnikforschung und der Infrastructure Studies. Im zweiten Teil kommen Methoden der Geschichtsschreibung, die durch medienarchäologische und diskursanalytische Ansätze angereichert sind, zum Einsatz. Die nun folgenden Abschnitte kartographieren die Sedimentierungen und verorten sie im Verhältnis zueinander.

Im Editorial zur ersten Ausgabe von *Computational Culture* zeichneten die Herausgeber 2011 eine Reihe von Gründen, Software zum Gegenstand der Untersuchung zu machen. Sie meldeten Zweifel in Bezug auf »die unterstellte, jedoch nicht wirklich nachgewiesene Überlegenheit von computabel optimierten Formen des Rechnens«<sup>v</sup> und »die stillschweigende Vorannahme, dass computerisierte Artefakte Dinge eleganter erledigen können« (Fuller u. a. 2011) an.<sup>vi</sup> Eine der Notwendigkeit der Untersuchung von Software sahen sie darin, die nicht formulierte oder nicht sichtbare »Artikulationsarbeit«, welche der Vorbereitung und dem Ingangsetzen rechnender Infrastruktur dient, sichtbar zu machen. Sie schlugen in der Folge vor, zu erforschen, wo und wie formal-materielle und soziokulturelle Prozesse ineinander greifen, welche Lücken sich zwischen den Softwarefigurationen und den spezifischen Einsatzkontexten auftun und welche Umgangsweisen die Nutzerinnen damit finden (ebd.).

Thomas Haigh bezeichnete Software als ein »fluides Konzept«, welches sich mit den Maschinengenerationen verändert habe. Historisch seien eine ganze Reihe von Softwarekonzepten zu verzeichnen. In den 1950ern stand die durch die Hersteller gelieferte Systemsoftware (Assembler, Compiler, Betriebssysteme) im Vordergrund, ergänzt durch spezifisch für das jeweilige Anwendungsgebiet erstellte Werkzeuge, Anwendungen und Dienste von Drittanbietern.

Die teils umfangreichen Eigenprogrammierungen der Nutzer von Computerhardware fielen in jener Zeit noch nicht unter den Begriff Software. Ab Mitte der 1960er Jahre schälte sich eine neue Umgangsweise heraus, denn Drittanbieter begannen nicht nur spezifische Problemlösungen anzubieten, sondern auch generelle Standardlösungen (Haigh 2002).

Software, und damit auch Datenbanken, wurden durch Wendy Chun als »unsichtbar« und »ephemer« beschrieben, woraus eine Reihe von Schwierigkeiten für ihre historische Erforschung folgte. Diese begannen damit, dass Software selten archiviert werde, dass die zugrundeliegende Hardware nur selten in noch lauffähiger Form vorhanden sei, und endeten noch längst nicht mit dem Umstand, dass eine Grenzziehung zwischen Software und Hardware grundsätzlich schwierig sei (Chun 2011, 3).

In diesem Spannungsfeld bewegt sich die vorliegende Arbeit. Denn Datenbanken dienen nicht nur der Vermittlung der Daten zwischen verschiedenen Programmen, sie sind als Datenbankmanagementsysteme selbst auch höchst anpassbare Standardfigurationen der Datenverwaltung. Und sie koordinieren Arbeitsprozesse. Haighs Unterscheidung in Programmierservices, Anpassungen und Eigenprogrammierungen im Unterschied zu ›Out of the Box‹-Standardsoftware wirft die Frage danach auf, welchen Schwerpunkt medientheoretische Erkundungen setzen. Auf PCs installierte ›Out of the Box‹-Software wie Microsoft Word, oder Adobe Photoshop, beziehungsweise die auf Smartphones installierten Apps und die Verwendung von Webanwendungen wie YouTube, Facebook und Instagram prägen, durch den tagtäglichen Umgang, unsere Vorstellung von Software und damit auch die entsprechende Forschungsarbeit. Dies hat zu einer teils einseitigen Schwerpunktsetzung der Forschung geführt. Problematisch ist daran, dass ein wesentlicher Anteil von Software auf der Beschreibungsebene ignoriert wird.<sup>20</sup>

Inzwischen sind auch die vergleichsweise weniger sichtbaren Softwarefigurationen, welche computervermittelte Arbeitsformen ermöglichen und begleiten, endlich in den Fokus der Medientheorie geraten. Mit Schüttpelz (2016, 1) argumentierend, lässt sich gerade anhand der Transaktion aufzeigen, dass das aktuelle technische Medium Datenbank auf vorhergehenden medialen Praktiken der Kooperation basiert. So werden durch den Autor Transaktionsbeispiele sowohl in Praktiken des Tabellierens in Fabriken zur Aufteilung von Arbeit als auch in komplexen mechanischen Informationssystemen für die Koordination im Dienstleistungssektor aufgezeigt. In der Datenbank wird, in einem kontingenten Prozess, die Transaktion von ihren Ursprüngen unabhängig und zu einer universalen Technik. Prozesse der Umwidmung, beziehungsweise Momente von Genesen zeichnet diese Studie daran anknüpfend anhand der Konstellation Datenbankmanagementsystem—Material und Enterprise Resource Planning (MRP/ERP) nach, indem sie im Tabellieren und der transaktionalen Kooperation genealogisch verankert werden. Diese Vorgehensweise gewinnt an Relevanz, folgt man der Fortsetzung von Schüttpelz' Argumentation, dernach alle vernetzten Alltagsmedien, also auch die heute stark wahrgenommenen Unterhaltungsmedien/›Soziale‹ Medien, »Anteile einer computergestützten kooperativen Arbeit« (ebd., 2) enthielten.

Tatsächlich sind Datenbanken heute überall anzutreffen, sei es, dass Menschen sich waschen und der Wasserverbrauch gemessen wird, dass sie miteinander telefonieren oder ein Computerspiel spielen, sich auf Dates verabreden, dass Lebensmittel oder Industrieroboter produziert werden, Einlass in Gebäude gewährt, oder Geld überwiesen wird. Das Problem der Medienforschung liegt darin, dass die

---

<sup>20</sup> Vgl. zum Beispiel Lev Manovichs grundlegende Studie *Software Takes Command* und darin spezifisch das Kapitel *Inside Photoshop* (Manovich 2013, 124–146), oder Till Heilmanns *Textverarbeitung* (Heilmann 2012). Zu den Social Media Studies stellvertretend *Networks Without a Cause – A Critique of Social Media* (Lovink 2011, 76–94), *Objects of Intense Feeling – The Case of the Twitter API* (Bucher 2013) und *Audience Labour on Social Media – Learning from Sponsored Stories* (Fisher 2015). Zum Feld der App-Studies siehe *Multi-Situated App Studies – Methods and Propositions* (Dieter u. a. 2019).



Untersuchung sozialer Medien oder des Internets größtenteils anknüpft an die Frage der Rezeption dessen, was die Bildschirme bereitstellen. Für Datenbanken ist die Frage der Rezeption weitestgehend irrelevant, sie sind Medien der Koordination, nicht der Rezeption und erfordern andere Beschreibungsweisen.

In dieser Perspektivnahme deutet sich an, inwiefern sich diese Untersuchung von der, durch Markus Burkhardt vorgelegten, grundlegenden medientheoretischen Monographie zu Datenbanken *Digitale Datenbanken – Eine Medientheorie im Zeitalter von Big Data* (2015) unterscheiden wird. Burkhardt entwickelte eine Medientopologie des Computers, die an die Idee der Oberflächen und Tiefendimension bei Niklas Luhmann anschließt: »Vor allem aber ändert der Computer, verglichen mit dem, was in der Tradition über Religion und Kunst definiert war, das Verhältnis von (zugänglicher) Oberfläche und Tiefe. [...] Die Oberfläche ist jetzt der Bildschirm [...], die Tiefe dagegen die unsichtbare Maschine, die heute in der Lage ist, sich selbst von Moment zu Moment umzukonstruieren, zum Beispiel in Reaktion auf Benutzung« (Luhmann 1998, 1:304).<sup>21</sup> Jedoch sei in dieser Oberflächen-Tiefen-Konstellation nicht länger jene klassische Differenz auszumachen, die unter der sichtbaren Oberfläche (z.B. eines Kunstwerkes) ein Telos der Tiefenerkenntnis erwarten ließe. »In der Tiefe digitaler Medientechnologien liegt keine wie auch immer geartete Wahrheit, sondern eine unsichtbare Maschine, die Signale gemäß der ihr gegebenen Befehle verarbeitet« (Burkhardt 2015, 81).<sup>22</sup> Die Oberfläche sei nicht länger Zeichen einer verborgenen Wahrheit, die es zu heben gelte, sondern vielmehr jene Ebene, auf der die Operationen von den Nutzern in Gang gesetzt werden, die in die unsichtbaren maschinellen Tiefen reichen. Die Vermittlung in die maschinellen Tiefen erfolge durch Befehle und Datenhaltung.<sup>23</sup> Burkhardt plädiert dafür, die phänomenale und semantische Ebene der Oberfläche, als auch die Techno-Logik der Tiefendimension, in ihrer Vermittlung durch Software, in den Blick zu nehmen.

Daran knüpft die vorliegende Studie an und erweitert sie. Sie beobachtet eingehender als Burkhardt die Emergenz und die Entstehungsbedingungen des Konzepts ›relationale Datenbank‹ aus den langen Operationsketten des Rechnens und Denkens in Mengenalgebra, des Tabellierens und der Transaktionalisierung – von der Datenbank zu den Datenbankpraktiken. Damit soll es gelingen, die relationale Datenbank deutlicher als bisher, als eingebettet in Softwareanwendungen und -umgebungen des Verarbeitens von Informationen (als Formation) zu kennzeichnen und die Wechselwirkungen mit diesen anderen Anwendungen – MRP, ERP, CRM usw. – so zu markieren, dass jene als verflochten mit Datenbanken erklärt werden. Es wird also stärker auf den alltäglichen Umgang mit Daten und Informationen in verwaltenden Organisationsstrukturen abgehoben und weniger stark eine Medientheorie der Datenbank, wie durch Burkhardt, entworfen.

---

<sup>21</sup> Luhmann äußert sich auf nur wenigen Seiten überhaupt zur medialen Konstellation des Computers. Burkhardt baut diese Skizze daran anschließend plausibel für digitale Datenbanken aus (Burkhardt 2015, 73–116).

<sup>22</sup> Entwickler heutiger Deep Learning und Mustererkennungssoftware, die unter dem Begriff »Künstliche Intelligenz« bekannt ist, erklären, dass sie nicht genau sagen könnten, was in den Tiefen ihrer algorithmischen Konstrukte vor sich geht, da zu viele Komponenten miteinander interagieren. KI wird daher anhand ihres Inputs und ihres Outputs bewertet (vgl. auch Jonas/Kording 2017).

<sup>23</sup> »Kommandozeile und GUI stehen nicht einerseits für den unmittelbaren Umgang mit der Technik und andererseits für die Mittelbarkeit von Interfaces. Beide sind gleichermaßen Oberflächen, die auf unterschiedliche Weise den Umgang mit der unsichtbaren Maschine strukturieren« (Burkhardt 2015, 84). Daraus folgt: »Software bedingt, welche Befehle zur Verfügung stehen, um welche Oberfläche mit welcher Tiefe in Verbindung zu bringen« (ebd. 87). Das Luhmannsche Konzept des Befehls verstärkt allerdings die Wahrnehmung prozessualer Abläufe und ignoriert die Bedeutung von Daten in der Vermittlung zwischen Tiefe und Oberfläche und ist daher zu erweitern.

Die Software Studies sind weit davon entfernt, ein vereinheitlichtes theoretisches und methodisches Programm zur Verfügung zu stellen. So überschneiden sich anthropologische, diskursanalytische und archäologische Zugangsweisen, die zum Teil angrenzend und überlagernd in eigenen wissenschaftlichen Abteilungen, wie den Infrastructure Studies und der Kulturtechnikforschung, gewichtet sind. Von dieser Schnittmenge ausgehend, sollen im Folgenden eine Übersicht über die wesentlichen Theorien und Methoden, die für die vorliegende Arbeit grundlegend sind, gegeben werden. Dabei spannen die Kulturtechnikforschung und die Infrastructure Studies einen Bogen auf, denn »die Beschreibung beider Konzepte bezieht sich auf Materialität, Verflechtungen und Prozessualität« (Schabacher 2020, Fol 1, pre-print).<sup>vii</sup> Dieser Bogen sei nicht unmittelbar offensichtlich, so Schabacher, da sich Kulturtechniken scheinbar eher auf Praktiken und Infrastrukturen als auf Materialitäten bezögen. Im Folgenden gilt es daher darzustellen, inwiefern beide Forschungsansätze zu einer Klärung im Feld der Software Studies beitragen können.

### 1.2.1 Kulturtechnikforschung

»Karteikarten, Schreibutensilien, Schreibmaschinen, Diskurs-Operatoren (inklusive Anführungszeichen), pädagogische Medien, wie Wandtafeln, verschiedene nicht-klassifizierbare Medien, wie Phonograph oder Briefmarken, Instrumente, wie das Klavier und Disziplinierungstechniken, wie die Alphabetisierung« (Siegert 2015, 2).<sup>viii</sup>

Datenbanken sind Kulturtechniken der Bürokratie. Die Untersuchung von Medienkulturtechniken fokussiert nicht wie andere Forschungsansätze auf die Inhalte oder Semantiken von Medien und deren Interpretation, sondern widmet sich der Frage des Umgangs mit Dokumenten, Materialien und Symbolen. Sie untersucht die medialen Bedingungen von Wissensproduktion selbst. Sie nimmt die Operationen, die mit einem bestimmten Medium bzw. einem Medienverbund möglich sind, in den Blick. Sie fragt, welche spezifische Medien sich mit anderen Medien im Gebrauch verbinden. Damit verschiebt sich der Forschungsfokus der Medientheorie von den Öffentlichen Medien zu den Medien der Kooperation, wie in diesem Fall: Datenbanken.

Datenbanken als Kulturtechnik werden hier unter den Aspekten ihrer Erlernbarkeit, der Naturalisierung und Unsichtbarkeit und der benötigten Fähigkeiten untersucht. Sie werden als Medienkulturtechnik erforscht in Bezug auf Materialität, Operationalität und Praxeologien, Medienvariantologien und die Exteriorisierungsthese.

Erstens, »Erlernbarkeit«: Kulturtechniken sind erlernbar und müssen eingeübt werden. Dies kann durch Anschauung und Nachvollzug erfolgen, in informellen Situationen oder auch formal. Beispielhaft steht dafür das Lesen-Lernen von Tabellen, welches die Tochter des Verfassers im zweiten bis vierten Schuljahr ebenso einübte, wie im Übergang vom 17. Jahrhundert zum 18. Jahrhundert König Ludwig, der XIV., der durch seinen Finanzminister Colbert das Lesen doppelter Buchhaltung erlernte (siehe Abschnitt 2.2.2. *Tabellarische Praktiken*).

Zweitens, »Naturalisierung und Unsichtbarkeit«: Der kulturtechnische Vollzug wird mit seiner Aneignung allerdings unbewusst. Seine Verwendung wird naturalisiert und damit als Kulturtechnik an sich

unsichtbar. In dieser ›Unsichtbarkeit‹ überschneiden sich Kulturtechnik und Infrastruktur, denn beide sind in den Alltag eingebettet. Diese Überschneidung beobachtet auch Gabriele Schabacher, wenn sie in dem Text *Medium Infrastruktur – Trajektorien soziotechnischer Netzwerke in der ANT* verschiedene Dimensionen der Unsichtbarkeit von Infrastrukturen kennzeichnet: Sie trete auf visueller Ebene (das Verstecken von Infrastrukturen), auf ästhetischer Ebene (Lärm, Schmutz, et cetera) und auf epistemologischer Ebene (fehlender Zugang zu Dokumentation bzw. zum ›Inneren‹ von Infrastrukturen) auf. In *Waiting – Cultural Techniques, Media, and Infrastructures* (Schabacher 2020) weist Schabacher noch einmal aus anderer Perspektive auf die ›Unsichtbarkeiten‹ von Kulturtechniken hin, und zwar in Bezug auf »unsichtbare Arbeit« (Star/Strauss), »implizites Wissen« (Polanyi) und »habituelle Routinen« (Bowker/Star). Wenn die vorliegende Arbeit beide Dimensionen aufruft, Kulturtechniken und Infrastrukturen, so mit dem Ziel, der Unsichtbarkeit durch präzise Beschreibung zu begegnen.

Drittens, ›Fähigkeiten‹: Hervorzuheben bleibt, dass aus Kulturtechniken abgeleitete Technisierungen wie die relationale Datenbank immer auch der Fertigkeiten bedürfen, denn »wir müssen technische Prozesse nicht als Ergebnisse von Intelligenz, sondern als auf Praktiken basierenden Fähigkeiten in den Blick nehmen« (Ingold 2001, 29)<sup>ix</sup>. Technologie stelle, so Ingold, eine Kodifizierung von Fähigkeiten (›Skill‹) dar, welche befolgt oder unterlaufen werden könne. Immer jedoch transformiere die Emergenz einer Technologie das Feld der Praktiken. Die benötigten Fertigkeiten im Umgang mit Datenbankmanagementsystem entziehen sich aufgrund der Komplexität und aufgrund verschiedener Nutzungsweisen, z.B. als Programmierer oder Endanwender, der vollständigen Beschreibbarkeit. Wenn daher das Tabellieren, die Abwicklung von Transaktionen oder die Auseinandersetzung mit den Vorgehensweisen des Material Resource Planning jeweils in eigenen Kapiteln fokussiert werden, so dient die Untersuchung dieser Fertigkeiten vor allem dazu, die heute verfügbare relationale Datenbankfiguration in den kontingenten Transformationsprozessen zu beschreiben, aus denen sie sich herausgeschält haben.

Viertens, ›Materialität, Operationalität und Praxeologien‹: Die vorliegende Studie befragt die historischen Quellen in Bezug auf die Operationalitäten und ihre Verkettungen (vgl. Siegert 2015, 10) einerseits und die damit verbundene Praxeologien (Schüttpelz 2016, 4f.) andererseits. Dies betrifft zum Beispiel die Beschreibung, wie sich die Operationalisierung von der Tabelle zur Datenbank hindurchzieht und sich in der Datenbank wandelt. Spezifische Operationalitäten gehen einher mit spezifischen Praktiken. Daraus folgt, dass die Fertigkeiten, welche zum Bearbeiten einer Tabelle, zur Abfrage einer Datenbank, oder zum erfolgreichen Verwenden von Enterprise Resource Management Software beitragen, hier historiographisch aus den in Technologie verkörperten »Kodifizierungen von Fähigkeit« (Ingold 2001, 30) rekonstruiert werden. Mit Seifert ist darauf zu bestehen, dass Praktiken immer auch Körperpraktiken sind und »keine rein ›geistigen‹ Operationen, die immateriell, funktionalistisch und abstrakt sind, sondern« im Körperlichen verankert »vielmehr auf ein Moment von Widerständigkeit und Alterität« verweisen (Seifert 2019, 111). Diese Körperpraktiken, zum Beispiel des Tabellierens, bedürfen spezifischer tabellarischer Materialitäten. Das eine lässt Rückschlüsse auf das andere zu. Es ist daher eine konzeptuelle Differenz einzuziehen: Die Kulturtechnikforschung konzentriert sich auf Operationen, um eine de-anthropozentrische Beschreibungsebene zu stärken, und untersucht daher Dispositive und Materialitäten. Die Praxeologien orientieren sich stärker an sozialwissenschaftlichen Handlungstheorien und untersuchen das ›Wie‹ des Umgangs mit Medien, aber auch die Vorbedingungen von Mediatisierung. Wie gerade Datenbanken dadurch zum Funktionieren gebracht werden, dass bestimmte in Software fixierte Operationen menschlicherseits vermittelt werden, wird im Kapitel 2.3.

*Koordination – Transaktionen* deutlich gemacht und in Abschnitt 3.2.3 *Dresden, Guben, Moskau, Rostock: Sachgebietsorientierte Programmiersysteme (SOPS)* detailreich belegt.

Fünftens, ›Variantologie‹: Was bedeutet die in der vorliegenden Studie vorgenommen Beobachtung von Datenbank-Emergenzen in Ost und West in den Zeiten des Kalten Krieges für eine kulturtechnische-praxeologische Untersuchung? Anthropologen haben lokal verschiedene Ausprägungen von Kulturtechniken und den damit verbundenen Fähigkeiten beobachtet (Widlock 2015, 45–47). Diese kontextuelle Differenz, welche man auch für den Osten und Westen vermuten könnte, entfaltet sich, wie wir sehen werden, in Bezug auf Datenbanken, Tabellen, Computer und mathematisch-logische Operationen in weitaus geringerem Maße, als zunächst anzunehmen. Die übergreifende Formation der Industrialisierung, so eine These, der im Detail nachzugehen ist, führt zwar zu lokal verschiedenen Ausprägungen der Nutzungen von Datenbanken. Doch sind an die Operationen gebundene Fähigkeiten in hohem Maße in andere Kontexte übertragbar, wie der Übergang des DDR-Personals der Datenbankentwicklung vom Osten in den Westen der wiedervereinigten Bundesrepublik Deutschland nach 1990 zu zeigen vermag. Trotzdem ist es wichtig, den Tiefenschichten verschiedener Genealogien von Datenbankmanagementsystemen nachzugehen, denn deren Variantologie vermag es, als Verkomplizierung naiver Auffassungen von ›Datenbanken‹ singulären Erzählungen entgegen zu wirken.

Sechstens, ›Exteriorisierung‹: Horst Bredekamp und Sybille Krämer kennzeichneten Kulturtechniken als Exteriorisierung (Krämer/Brekamp 2003, 18). Demnach wird die Operationalisierung von Symbolen erst durch Auf-Gezeichnetes erlangt. Dieses folgt bestimmten Regeln, welche eine spezifische Einschränkung des Hantierens mit den noetischen Gegenständen erzeugen. Der Umgang mit Informationsobjekten und in der Folge der Denkprozesse, wird vereinfacht, indem Kontingenzen und Unschärfen abgespalten werden. Es ist ein Vorteil, der gleichzeitig zum Nachteil gerinnt, denn im Zuge der Entlastung werden bestimmte Aspekte der Realität abgespalten und ignoriert. Dies materialisiert sich im Informationsmodell in Datenbankmanagementsystemen, welches Realität zugunsten von Daten filtert und nicht-computable Aspekte abspaltet.<sup>24</sup>

Die These der Exteriorisierung läuft allerdings Gefahr, wenn sie als akkumulierender Prozess aufgefasst wird. Schüttpelz kritisiert dies am Beispiel der Medienanthropologie Leroi Gourhans aus den 1950er Jahren, da jene auf eine linear zunehmende Exteriorisierung von Handtätigkeiten in Werkzeugtätigkeiten und Entlastung der Geistestätigkeiten durch Speicherung abziele (Schüttpelz 2006, 7). Tatsächlich ist man verführt, der Gourhanschen These unmittelbar zu folgen, wenn man die Datenbank als Speichertechnik und damit Auslagerung des Gemarkten fassen würde. So sehr Technologien und Kulturtechniken des Umgangs mit Datenbanken als Exteriorisierung erscheinen mögen, mit Widlock (2015, 50, 58) bleibt festzuhalten, dass es sich nicht um eine zunehmende Entfremdung, sondern um ein »re-embedding« (ebd.) der Fähigkeiten und Operationalisierungen in andere Zusammenhänge handelt. Der kulturtechnische Umgang bleibt immer an den Körper gebunden, an Subjektivierung und Affekt.

Die Überlegungen zu Erlernbarkeit, Naturalisierung & Unsichtbarkeit, Fertigkeit, Operationalität und Praxeologie, Variantologie und Exteriorisierung zeigen an, auf welche Weise die Kapitel 2.1 *Formalisierung – Strukturierte Daten in der relationalen Algebra*, 2.2 *Operationalisierung – Tabellieren* und

---

<sup>24</sup> Es werden auch computable Aspekte abgespalten, wenn es nach der jeweiligen Effizienzlogik überflüssig erscheint, diese als Daten zu erheben. Im Big-Data Approach gibt es im Prinzip keine überflüssigen Daten. Hier wird möglichst viel erhoben, ausgehend von dem Versprechen, die gesammelten Aggregate in Zukunft brauchbar machen zu können.

2.3. *Koordination – Transaktionen* anders als der historische, dritte Teil der Studie positioniert sind. Sie verdeutlichen, in welchem Maße die Operationalisierungen der Menge als relationale Algebra, der Tabelle und der Transaktion in Datenbankmanagementsystemen zu Technologie geronnen sind. Damit ist beabsichtigt, den gesamten Text zu repositionieren. Er rekonstruiert nicht allein Genealogien der relationalen Datenbank, sondern sucht außerdem jene vorhergehende Operationalisierung von Informationen als Formation auf, die nach und nach spezifisch für relationale Datenbanken geworden sind. Die grundlegende Bedeutung der Datenbank erscheint dann nicht mehr als Folge einer technologischen Revolution, sondern als Teil soziotechnischer Kontinuitäten. Sie wird einerseits im Höhepunkt der Industrialisierung verankert und verweist andererseits mit der Tabelle auf die Anfänge der Schriftlichkeit.

*Zusammenfassend:* Beobachtungen von kulturtechnischen Formationen rekurren weniger auf den Inhalt von, als auf den Umgang mit Dokumenten, Materialien und Symbolen. Als Leitmotiv der Untersuchung soll die genealogisch begründbare Darstellung von Operationsketten dienen, welche der relationalen Datenbank vorhergehen. Dies vermag es, für die Software-Studies Argumente zur Verfügung zu stellen, die das Gewordensein heutiger medialer Figurationen aus der Perspektive von medialen Praktiken präzisiert.

## 1.2.2 Infrastructure Studies

»Die Herkunft von Datenbankmanagementsystemen kann auf die Entwicklung von Datendefinitionen, Report-Generatoren, und die Command-and-Control Systeme der 1950er zurückgeführt werden – eine Zeit in der Computer erstmals für die Verarbeitung ökonomischer Daten verwendet wurden« (Fry/Sibley 1976, 19).<sup>x</sup>

Datenbanken sind Infrastruktur.<sup>25</sup> Sie ermöglichen das Funktionieren zahlreicher anderer Infrastrukturen, Dienste und Prozesse. Sie sind in Schichten von Betriebssystemen, Softwareanwendungen und Hardware eingeordnet. Im Sinne der Infrastructure Studies haben Bowker/Baker/Millerand/Ribes im Anschluss an Ruhleder/Star (1995) das Eingebettet-Sein von Infrastruktur betont: »Typischerweise existiert Infrastruktur im Hintergrund, sie ist unsichtbar und ihr Funktionieren wird allgemein vorausgesetzt« (Bowker u. a. 2010, 98)<sup>xi</sup>. Die Nicht-Sichtbarkeit von Datenbanken ist demnach die unmittelbare Folge ihrer tiefen Einbettung. Eine Wiedergewinnung von Sichtbarkeit stellt insofern eine Hauptabsicht der vorliegenden Studie dar. Daraus folgt die Frage, welche Mittel die Infrastructure Studies für dieses Vorhaben zur Verfügung stellen.

Erstens, ›Infrastrukturelle Inversion‹: Dem Entzug von Wahrnehmbarkeit lässt sich eine Strategie der Sichtbarmachung entgegen setzen, die mit Goeffrey Bowker als ›infrastrukturelle Inversion‹ gefasst werden kann. Von den zugänglichen Enden einer Infrastruktur ausgehend wird im Sinne einer

---

<sup>25</sup> Schabacher verortet Infrastruktur als historisches Konzept »im Zusammenhang mit den sich im Zuge der Industrialisierung etablierenden Verkehrs- und Kommunikationsinfrastrukturen des 19. Jahrhunderts und den damit einhergehenden kolonialen Bestrebungen etwa der ›Erschliessung‹ Afrikas« (Schabacher 2013, 134).

Umkehrung angestrebt, Teile der Infrastruktur als diskursives Feld zu rekonstruieren (vgl. Parks 2010).<sup>26</sup> Der Methode ist es wichtig, Infrastruktur nicht allein als ein technisch-materielles Set zu verstehen, sondern als eine nicht-statische Figuration. Diese wird einerseits durch die politisch-administrative Aushandlung von Standards und Strukturen (Schabacher 2013, 141–143), und andererseits durch die sozialen Interaktionen der Nutzer, Manager und Maintainer geformt (Bowker u. a. 2010, 99). Die methodischen Grenzen liegen im zugänglichen Material begründet. Ein Teil der Infrastrukturen wird notwendigerweise unsichtbar bleiben, unter anderem da beispielsweise kein zentrales Betriebsarchiv von Robotron, aber auch nicht der Firma SAP existiert. Es wurde versucht, dies durch Interviews auszugleichen, welche in der Lage sind, auf andere Weise Auskunft zu geben als Betriebsanleitungen, Zeitungsartikel oder die Beobachtung technischer Setups im Computermuseum.

Schabacher weist auf die Probleme hin, Material prozessual zu denken, »also von einer grundlegend nicht stillzustellenden Dimension der Arbeit an und von Infrastrukturen auszugehen« und kommt zu dem Schluss, dass »stets nur lokale Visualitäten und begrenzte Accounts« (Schabacher 2013, 148) beschreibbar sind. Daraus folgt, dass das Mapping von Infrastrukturen zwangsläufig partiell und unvollständig bleibt. Einen Gesamtüberblick zu verschaffen, wäre ein mutwillig falsches Versprechen. Umso wichtiger ist es, die Partikularitäten derart zu verdichten, dass sie wie ein Rhizom in die unweigerlichen Leerräume vordringen und Ausläufer bilden, ohne die Leerstellen zu überlagern oder zu verdecken.

Zweitens, »Infrastruktur als Medium«: Nimmt man Ned Rossiters Diktum »Wenn Infrastruktur Welten hervorruft, dann werden sie durch Software koordiniert« (Rossiter 2016, xv)<sup>xii</sup> ernst, dann sind Datenbanken als Warenlager (Domman 2012), aufgrund ihrer relativen Unsichtbarkeit als Gray Media (Fuller/Goffey 2012, 2), Logistical Media (Rossiter 2016), als Mitte und Mittel der Kooperation (Schüttpelz 2016, 6) und Koordination (Gießmann 2018, 98f.) zu kennzeichnen.

Die vorliegende Untersuchung widmet sich daher nicht allein den Aspekten der Speicherung, Berechnung und Übertragung in und mit Datenbanken, sondern den Genealogien, die hinter »der Banalität von Tabellenkalkulationen, Enterprise Resource Planning (ERP) Systemen und Softwareanwendungen« (Rossiter 2016, xvi),<sup>xiii</sup> im Verborgenen zu bleiben drohen. Daher ist der »Banalität« der Tabelle ein ausführliches Kapitel gewidmet: 2.2 *Operationalisierung – Tabellieren*. Das Kapitel 3.2 *Robotron und SAP: Sozialistisches und kapitalistisches Enterprise Resource Management im Vergleich* rekonstruiert eine osteuropäische und eine westliche Variantologie der Genealogie von Enterprise Resource Planning Software.

Zur Infrastructural Inversion gehört eine häufig kleinteilige Untersuchung, wie sie Fuller/Goffey für *Gray Media* aufrufen: »Das dezente Grau so vieler Arten von Medienpraktiken, von der Systemadministration zur Datenerhebung oder der Kontrolle und Verifikation aller Arten von Qualitäten und Attributen fordert zu einer misstrauischen Aufmerksamkeit auf, einer Kultivierung jener Sensibilität, die in der Lage ist, geringe Änderungen in den Nuancen, Hinweise auf Kontrast, wo Kontrastmangel ansonsten die Regel wäre, wahrzunehmen« (Fuller/Goffey 2012, 12).<sup>xiv</sup> Es lohnt sich daher, in großem Detail herauszuarbeiten, wie die DDR-Variante von IBMs Software Bills of Material Processor (BOMP) namens Bankspeicherung Technischer Informationen (BASTEI) von einer spezifischen Materialverwaltung zu einer generellen Datenbank mutierte und wie daran die infrastrukturellen Maintainer beteiligt waren. In dieser emergenten, abgewandelten Nutzung konnte BOMP/BASTEI universelle

---

<sup>26</sup> Infrastrukturelle Inversion kommt nicht allein in Textform zum Ausdruck. Sie ist auch Methode eines Teils der künstlerischen Praxis des Autors, welche inspiriert durch Lisa Parks infrastrukturelle Inversionen von Satelliteninfrastruktur (Parks 2010) und eigene *Database Derives* durchführte (Hunger 2015).

Datenverarbeitungsaufgaben abwickeln und wurde zum Vorbild für das spätere, netzwerk-orientierte Datenbanksystem/Robotron (DBS/R). Diese Umdeutung wird im Abschnitt 3.2.1.1 *Explosionen: Bankspeicherung Technischer Informationen (BASTEI) / Bill Of Materials Processor (BOMP)* detailliert dargestellt. Auch die im Abschnitt 3.2.2.2 *Einheitliche Primärdokumentation* diskutierte Einheitliche Primärdokumentation, sowie die folgenden Abschnitte zu Pflichtenheft und Datenerhebung folgen der, von Fuller/Goffey geforderten, nuancierten Aufmerksamkeit.

Drittens, »Infrastrukturelle Arbeit«: Rossiters *Logistical Media* eröffnen den Raum für eine intensive Befragung der, mit Hilfe von Datenbanken organisierten, Arbeit. Die bereits genannten Kapitel tragen zu diesem Feld bei, denn der Autor ist davon überzeugt, dass ihm bisher nicht genügend Aufmerksamkeit gewidmet wurde.<sup>27</sup> Stattdessen haben sich in der jüngsten Vergangenheit »Soziale« Medien Plattformen, wie beispielsweise Instagram, Facebook, Twitter im tagtäglichen Umgang mit dem Smartphone als offensichtlicher und eingängiger Untersuchungsgegenstand den Medienwissenschaften aufgedrängt. Anders ist dies bei den logistischen Medien. Logistical Media umgeben uns in mindestens gleichem, wenn nicht sogar größerem Maß, zum Beispiel als Verkehrslogistik, als Grundversorgung mit Wasser oder als Warenlogistik, welche dafür sorgt, dass die Supermärkte gefüllt sind. Angesichts dessen, ist ihre medientheoretische Erforschung mit weitaus größerer Dringlichkeit gegeben, als bisher geschehen. Ein wichtiges Anzeichen dafür ist die bisher ungeschriebene Frühgeschichte der SAP-Software, welche im Abschnitt 3.2.4 *Walldorf: SAP und die modulare Integration* erarbeitet wird, um diese der DDR-spezifischen modularen Problemlösungssoftware SOPS gegenüber zu stellen.

Mit Rossiters Analyse gilt es nun zu beobachten, wie die Vermessung der Welt in Echtzeit mittel Datenbanken ermöglicht wurde. Denn, so Rossiter: »Logistische Software wirkt als eine Technologie der Governance und Kontrolle, indem sie die Produktivität von Arbeit anhand von Echtzeit-KPIs [Key Performance Indicators – F.H.] misst. Die Produktion neuer Arbeitssubjektivitäten ist der Logistik innewohnend« (Rossiter 2016, 9)<sup>xv</sup>. Die neue Qualität liegt nicht so sehr in der Datenerhebung über Arbeitsvorgänge, dies sind die bekannten Methoden des Taylorismus, sondern in der Echtzeiterhebung und -auswertung. Diese Echtzeiterhebung betrifft nicht mehr allein Arbeitsvorgänge. Sie umfasst inzwischen auch viele andere Bereiche des Lebens, die automatisiert erfasst, zu Statistiken zusammengefasst und gegeneinander abgeglichen werden.<sup>28</sup> In relationalen Datenbankmanagementsystemen wurde diese ab Anfang der 1990er Jahre mit Hilfe des On-line Analytical Processing (OLAP) in Datawarehouses technisiert, siehe Abschnitt 2.3.7 *On-Line Analytical Processing (OLAP)*. Die angesprochenen neuen Subjektivitäten der Arbeit haben sich in einem längeren, emergenten Prozess seit den 1960er Jahren herausgeschält. Während Rossiters Beschreibung vorrangig in die Gegenwart investiert, soll die vorliegende Untersuchung stärker in die Vergangenheit zurück blicken, denn es ist zu vermuten, dass die neuen Subjektivitäten der Arbeit und des Arbeitens sich über lange Zeiträume und lokal unterschied-

---

<sup>27</sup> Zwar existieren durchaus einschlägige soziologische und techno-anthropologische Studien, wie beispielsweise *Das Projekt SAP – Zur Organisationssoziologie betriebswirtschaftlicher Standardsoftware* (Mormann 2016) oder *Organisation im soziotechnischen Gemenge – mediale Umschichtungen durch die Einführung von SAP* (Conrad 2017). Im Unterschied dazu sollen die operationalen Verwebungen mit relationalen Datenbanken in der vorliegenden Studie stärker als bisher positioniert werden, um damit Aussagen über die Genese relationaler Datenbankmanagementsysteme treffen zu können.

<sup>28</sup> Dies ist ein Hinweis darauf, dass sich der Status von Arbeit wandelt. Alles was datafiziert wird, wird der Arbeit unterworfen. Damit wären auch Tätigkeiten wie Waren- oder Medienkonsum als Audience Labor zum Feld der Arbeit zu zählen (vgl. Fisher 2015; Nixon 2015).

lich verteilt, entwickelt haben, wie beispielsweise die im Abschnitt 2.2.2.3 *Transaktionen – Plantafeln: Prozess- und Organisationstabellen* verhandelten Planungstabellen zeigen.<sup>29</sup>

Viertens, »Medien der Koordination«: Wenn mit Rossiter hier die Rede davon ist, dass Datenbankmanagementsoftware die Welt koordiniert, stellt sich die Frage nach dem Spezifischen der Koordination. Koordination sei, so Gießmann, selbst wenn es eine zentrale koordinierende Person gebe, weniger als ein Top-Down-Geschehen zu markieren, »denn als elementare Verständigung einer Praxisgemeinschaft über das jeweils angemessene Vorgehen in einem gegebenen organisatorischen Kontext« (Gießmann 2018, 101).<sup>30</sup> Schließlich müssten auch die Koordinierten den Prozess und seine Repräsentationen anerkennen und befördern. Diese Position bindet ihre Beobachtungen in eine Systematisierung der Praktiken infrastruktureller Medien ein, die eine medientheoretische Verschiebung durchführt: »Aus dem ›Speichern‹ würde ein ›Registrieren / Identifizieren‹ werden, mit dem die kontrollgesellschaftlichen Eskalationen und Datenobsessionen neuerer sozialer Medien beschrieben werden können. ›Übertragen‹ wäre weitaus besser fassbar, wenn man es als Frage der Handlungsverkettungen im ›Delegieren‹ zwischen menschlichen und nichtmenschlichen Agenten versteht. Und was im ›Bearbeiten‹ geleistet wird, bedarf der ›Koordination‹ von Körpern, Apparaten und Arbeitsteilung [...]« (ebd., 108). Für die vorliegende Arbeit wird sich dieser Ansatz als produktiv erweisen, denn relationale Datenbankmanagementsysteme automatisieren erstens das ›Registrieren/Identifizieren‹. Sie dienen zweitens dem ›Delegieren‹ besonders, indem das relationale Modell Informationen algebraisch für Maschinen und Menschen zur Verfügung stellt, als auch drittens das ›Koordinieren‹, welches vor allem in der Technisierung von Transaktionen eingebettet wurde. Die im Sozialen verortbaren Praktiken seien immer auch kategorial zu denken, denn »analytische Kategorien [stellen] ein Gegengewicht dar, mit dem Praktiken anders angeordnet und analysiert werden können, als es die Akteur\_innen selber tun würden« (ebd., 96). Indem die in dieser Studie betrachteten Infrastrukturen als Medien der Koordination, beziehungsweise der Kooperation (Schüttpelz/Gießmann), verhandelt werden, wird die Dringlichkeit der Untersuchung von Datenbankinfrastrukturen deutlicher.

### 1.2.3 Kooperative Figuration

Datenbanken sind Softwarefigurationen. Sie sind somit immer bereits verwoben und auf anderer Infrastruktur basierend, nie sind sie allein lauffähig. Immer stellen sie ein komplexes Konstrukt verschiedenster Komponenten dar. Wie einheitlich, wie systematisch ist diese Einheit, namens Datenbank? Welches Konzept wird dem Umstand der Verflochtenheit und des Eingebettet-Seins gerecht? Geklärt werden sollen diese Fragen anhand zweier Konzepte. Im Alltagssprachgebrauch des Computings ist die Rede vom ›System‹, doch erscheint dies für die vorliegende Untersuchung als unzureichend, weshalb das Konzept der ›Figuration‹ als Erweiterung vorgeschlagen wird.

---

<sup>29</sup> Die hier vorliegende Darstellung konzentriert sich auf den Zeitraum in der DDR. Für Westeuropa, und mit anderem theoretischen Zugang, wurden die sich verändernden Subjektivitäten unter anderem durch Maurizio Lazzarato in *Immaterielle Arbeit – Ästhetisierung der Politik und der Produktion unter den Bedingungen des Postfordismus* (1998) und Paolo Virno in *Grammatik der Multitude – Öffentlichkeit, Intellekt und Arbeit als Lebensformen* (2008) beschrieben.

<sup>30</sup> Schüttpelz initiierte den DFG-Sonderforschungsbereich *Medien der Kooperation* an der Universität Siegen, in dem diese theoretischen Positionen unter Mitarbeit von Gießmann und anderen entwickelt wurden. Eine Sammlung der Ergebnisse enthält der Band *Materialität der Kooperation* (Gießmann/Röhl/Trischer (Hrsg.) 2019). Eine Kritik an Schüttpelz' und Siegerts Konzept der Vorgängigkeit von Operationsketten entwickelt Heilmann in *Zur Vorgängigkeit der Operationskette in der Medienwissenschaft und bei Leroi-Gourhan* (Heilmann 2016).



Datenbankmanagementsysteme (DBMS) bringen es bereits im Namen mit sich: Das System. Die im Computergebrauch eingebürgerte Verwendung ist ingenieurtechnisch geprägt. Ist dieser Bedeutungsschwerpunkt für das oben formulierte Erkundungsinteresse ausreichend? Muss er gegebenenfalls aktualisiert oder gewendet werden? In mehreren Schritten ist diesen Fragen nachzugehen.

Erstens, ›historisches System-Konzept‹: In den Naturwissenschaften ist ›System‹ im Zuge der zunehmend Empirisierung ab Anfang des 18. Jahrhunderts wichtig geworden. Zentral für dieses Systemkonzept ist die Annahme von beobachtbaren Kausalitäten. »Wir definieren ein ›System‹ als eine Anzahl von in Wechselwirkung stehenden Elementen  $p_1, p_2 \dots p_n$ , charakterisiert durch quantitative Maße  $Q_1, Q_2 \dots Q_n$ « (Bertalanffy 1945, 32). Komplexe ›Systeme‹ konnten fortan über die Beschreibung ihrer kleinsten Einheiten und deren Beziehungen untereinander rekonstruiert werden. Modernes Wissen konstituierte sich nunmehr aus systematisch kombiniertem Wissen über die innere Konfiguration der Einzelteile und über das Verhältnis der systemischen Gesamtheit zum Außen. Im wissenschaftlichen Experiment konnte isoliert die Veränderung einzelner Variablen vorgenommen oder simuliert werden und die Gesamtbewegung des Systems beobachtet werden. Dieses modernistische Programm ist eng verflochten mit der Kybernetik der 1940er–1970er Jahre und wurde nicht nur auf Kontrollmechanismen und Rechenmaschinen, sondern auch auf zahlreiche andere Felder angewandt: Künstliche Intelligenz, Management- und Operationsforschung, Verhaltensforschung, soziale Systemtheorie und Autopoiesis. Zusätzlich, so argumentieren Neubert und Wiemer, verbinde ›Simulation‹ als Erkenntnismittel die Felder der Biologie, Ökologie und Systeme (Neubert/Wiemer 2014, 10f.).

Komplexitätsreduktion und Simulation, welche eine vielfältige innere Zusammensetzung als äußerlich abgeschlossen und abgegrenzt zur Umwelt erscheinen ließen, machten das System für die Ingenieurwissenschaften attraktiv. So wurden Systeme als beschreibbar durch Elemente, Beziehungen der Elemente und Eigenschaften der Elemente aufgefasst. Dies führte zu Begriffen wie Computersystem, Datenbankmanagementsystem, Informationssystem oder Betriebssystem, mit dem Vorzug, dass diese in internen und öffentlichen Diskursen als Kausal-Gesamtheit kommunizierbar wurden.

Für die Medientheorie erwiesen sich zwei Schwerpunkte als folgenreich, erstens die Untersuchungen sozialer Systeme durch Niklas Luhmann (1987 und weitere), und zweitens die Forschungen zur Kybernetik, am prominentesten hier Norbert Wiener (1948).

Zweitens, ›Grenzen des System-Konzepts‹: Mit dem Konzept ›System‹ trat eine bedeutende Verschiebung ein. Es erlaubte nicht nur die empirischen Beobachtung von Wirklichkeit, sondern ermöglichte vielmehr die Konstruktion von Wirklichkeit. Problematisch wird die ingenieurstechnische Verwendung daher genau, wenn es um die Definiertheit der Grenze und des Innen und Außen geht. Unschärfen, Zufälligkeiten oder unerwartete Komplexität treten in dieser Sichtweise als zu bewältigende und bewältigbare Probleme auf, die, so scheint es, ihrer Lösung harren – eine Sichtweise, die inzwischen mit dem eigenen Begriff *Solutionism* belegt ist.<sup>31</sup> Die Grenze, welche ein Innen und ein Außen suggeriert, enthebt Ingenieure scheinbar von der Verantwortung gegenüber dem ›Außen‹ des Systems. Die Rede vom System erlaubt es, Unschärfen als unerfreulich zu markieren, nach außerhalb zu delegieren und zu ignorieren. Sie erlaubt es, Exaktheit zu suggerieren und befördert mechanistische Auffassungen von Realität. So wird Realität als all das begriffen, was systematisiert werden kann und das

---

<sup>31</sup> Morozov verdeutlicht, dass Solutionism einhergeht mit einer Ignoranz gegenüber ethischen und sozialen Fragen (Vgl. Morozov 2013, 351f.).

›Andere‹ abgespalten. Dies erzeugt Einschlüsse und Ausschlüsse während die Verantwortung für die Entscheidungen, die in diesem Zuge getroffen wurden, durch die Akteure negiert werden können.

Drittens, ›Datenbank-Systeme‹: Die vermeintliche Stabilität, die einem System zugeschrieben wird, besteht im Datenbankmanagementsystem ausgerechnet nicht, denn es kombiniert Stabilität und Instabilität. Indexe, um ein Beispiel anzuführen, welche ein schnelleres Durchsuchen erlauben, sind nur solange stabil, bis sie erneut aus den geänderten Daten aufgebaut werden. Ähnlich verhält es sich mit Zugriffsrechten und Transaktionszuständen, welche Datenbanken beständig instabil–stabil halten. Das Konzept ›System‹ in Datenbanken resoniert mit der Closed-World-Assumption der Informationsmodelle relationaler Datenbanken: Alles, was nicht als ›wahr‹ bekannt ist, ist als ›falsch‹ anzusehen (Date 2007, 96–115, 113f.). Im Unterschied dazu stünde die Open-World-Assumption: Alles was nicht als ›wahr‹ bekannt ist, ist ›unbekannt‹ und nicht notwendigerweise falsch.<sup>32</sup> Die computable Wahr/Falsch-Logik der Closed-World-Assumption zieht ein Innen und ein Außen nach sich, dessen Grenze im Informationsmodell gezogen wird, und nur aus dieser Perspektive macht es Sinn, das Datenbankmanagementsystem als System zu fassen. Es soll allerdings der vorliegende Text andere Perspektivnahmen, als allein die ingenieurstechnische, vornehmen.

Bleibt die Aufgabe sich festzulegen, wie mit dieser epistemologischen Barriere umgegangen werden kann. Es soll für diese Studie folgende Unterscheidung getroffen werden: In Zusammenhängen mit historischen Quellen, in denen das Konzept ›System‹ aufgerufen wird und im Kontext feststehender ingenieurstechnischer Fachtermini wie ›Datenbankmanagementsystem‹, wird dieser weiter verwendet, um genau diesen Umstand mit einer Markierung zu versehen.

Jenseits von System steht eine Reihe anderer Konzepte wie Netzwerke, Ökologien,<sup>33</sup> Digitale Milieux,<sup>34</sup> oder Assemblagen<sup>35</sup> mit jeweils eigenen Problemen zur Verfügung. Die vorliegende Studie wird in der

---

<sup>32</sup> Diese Vorannahme wird beispielsweise für die Sammlungen des Semantic Web vorgeschlagen. Wobei eine derartige Sicht unter anderen durch Date kritisch gesehen wird. Er argumentiert, dass die *Open World Assumption* Wahr-Falsch-Unbekannt zu einer ternären Logik führen würde, was wiederum für die binär argumentierenden Prädikatenlogik der relationalen Algebra zu erhöhten Schwierigkeiten der Implementation führen würde (ebd.).

<sup>33</sup> Karen Ruhleder und Susan Leigh Star haben 1996 in Anlehnung an Gregory Batesons' 1978 ethnographischen Ansatz *Steps towards an Ecology of Mind* vorgeschlagen, Informationsinfrastrukturen als Ökologie (»Ecology«) zu fassen (Star/Ruhleder 1996). Problematisch jedoch ist, dass mit Ökologie unweigerlich Arthur Tansleys Ökosystem aufgerufen wird, welches sich dadurch auszeichnet, dass es sich idealerweise in Balance befindet und nicht aus den Fugen gerät (Tansley 1935, 300). Damit wären wir erneut zurück gekehrt in das konzeptuelle Feld des Systems und andererseits bei der impliziten und ihrerseits problematischen kybernetischen Vorstellung, dass ein Equilibrium, ein ›natürliches‹ dynamisches Gleichgewicht, der anstrebenswerte Zustand eines Ökosystems, bzw. von kybernetischen Systeme an sich sei (Kritisch dazu 2011; Brine/Poovey In: Gitelman 2013, 61–76; Hörl 2016, 40f.).

<sup>34</sup> Der medienphilosophische Konzept digitaler Milieux (Hui 2016, 26, 57f.) ergibt Sinn aus Yuk Huis Zentralstellung digitaler Objekte (Daten), die in Millieux eingebettet sind.

<sup>35</sup> Für die Medientheorie ist der Ansatz, die Hierarchisierung zwischen Künstlichem und Echtem und zwischen menschlichen und nicht-menschlichen Akteuren aufzulösen, in den letzten Jahrzehnten äußerst fruchtbar gewesen. Die Kritik eines anthropozentrischen Bias der Sozialwissenschaften und deren Überhöhung menschlicher Handlungsmacht zählte ab Ende der 1980er Jahre zu den innovativen empirischen Methoden, welche in die Akteur-Netzwerk-Theorie mündeten. Es bleibt jedoch prekär, was in horizontalisierenden Relationen das Humane ausmacht, wenn es allein als Entität verhandelt wird. Zumindest, was Datenbanksoftware und -anwendungen angeht, kann festgehalten werden, dass sie ihre Existenz, unabhängig davon, wie stark ihre Operationen als Handlungen eigenständiger Akteure gelesen werden können, dem menschlichen Streben zu verdanken ist, sich der stupiden Arbeit des Sortierens und Verwaltens von Daten zu entledigen und Arbeit zu automatisieren (vgl. Schüttpelz 2006; White 2013; Brennan 2016; Wylie 2016; Gießmann 2018). Neben diesen grundsätzlichen Erwägungen wurde kürzlich von Dieter et. al. in Frage gestellt, ob die von Venturini/Jacomy/Meunier/Latour 2017 erneuerte Prämisse

Unterscheidung zum ›System‹ das Konzept der ›Figuration‹ aufrufen. Wenn man den aktuellen Debatten folgt, fällt auf, dass das Konzept der Figuration zunehmend in medialen und kommunikativen Kontexten Verwendung findet. Es erlaubt, zwei Konzepte aufeinander zu beziehen: das sozialanalytische Konzept des Netzwerks (Castells 2010) einerseits und das kulturanalytische Konzept der Assemblage/ des Gefüges andererseits (Deleuze/Guattari 1980, 124f.; De Landa 2006). Es soll in zwei Schritten auf seine spezifische Eignung hin entfaltet werden, um Datenbankpraktiken zu fassen. In einem ersten Schritt ist daher zuerst die Figuration darzustellen, ein Begriff, welcher ursprünglich der Soziologie entstammt und teils in die empirisch-quantitativen Kommunikationswissenschaften übernommen wurde. In einem zweiten Schritt wird es möglich, das Konzept für Datenbankpraktiken neu zu fassen, indem es als kooperative Figuration repositioniert wird.

Erstens, ›Figurationswandel und Rahmenanalyse‹: Die Figurationssoziologie, entwickelt als Prozesssoziologie in Abgrenzung zu statischeren Konzepten, bezog sich ursprünglich auf menschliche Beziehungsgeflechte. Die Figuration fragt danach, auf welche Weise die Elemente in eine Konstellation geraten sind, danach, wie der Prozess des Figurierens, der »Figurationswandel« (ebd., 14) verlaufen ist. Der Rahmen (Goffman 1974, 21–39) setzt in einer Figuration eine Ausrichtung der Handlungsmöglichkeiten, wobei sich mehrere Rahmen überlagern können. Dies bildet auch sich wandelnde Handlungsmöglichkeiten ab. Rahmen, die einander überlagern, ermöglichen andere Beschreibungsebenen, als beispielsweise Netzwerke, Systeme oder Assemblagen und zwar durch das Register analytischer Trennungen. Kuipers (2018, 429) verdeutlicht dieses Potenzial des Figurationskonzepts: Auch wenn in Figurationen weiterhin relationale Beschreibungsweisen verwendet werden, so erschöpfen sie sich nicht in flachen Ontologien oder potentiell endlosen Netzwerken, sondern verwirklichen die Potenziale analytischer Fokussierungen, indem verschiedene Rahmen, deren Entstehungsprozesse und Hierarchien nachgezeichnet werden.

Zweitens, ›Kommunikative Medien und Medien der Kooperation‹: Mit dem Diktum, dass »alle Medien kooperativ erarbeitete Kooperationsbedingungen und als solche entstanden« sind, weist Schüttpelz (2016, 1) darauf hin, dass Medientheorie und Sozialtheorie gemeinsam zu denken sind und zwar *neu* aus der Perspektive der Kooperation.<sup>36</sup> Derzeit sei zu konstatieren, dass die einflussreichen Medien- und Kommunikationstheorien kaum darauf angelegt sind, infrastrukturelle Register der Kooperation zu beschreiben. Zu dieser Leerstelle trug in der Vergangenheit die Konzentration der Medienforschung auf Massenmedien und Unterhaltungsmedien bei.

›Soziale‹ Medien stehen zwar als Kommunikationsmedien im Zentrum der Digitalisierungsdebatten der Medienwissenschaften, allerdings kommen sie ohne Datenbankpraktiken, überhaupt nicht aus. Für das digitale Zeitalter gilt: Ohne Medien der Koordination, keine Medien der Kommunikation. Die hier vorgelegte Studie leistet in diesem Sinne Grundlagenarbeit. Sie verdeutlicht, dass die Medien der Kooperation, welche historisch gesehen vor allem der Organisation von Arbeit dienten, alltagsfigurierend sind (vgl. auch ebd., 15).

Eine neue Erkenntnis ist dies nicht. So ließen sich die ausführlichen Beschreibungen der Fabrikarbeit im England des 18. und 19. Jahrhunderts durch die politischen Ökonomen wie Smith, Taylor, Ricardo, Marx und Engels als Medientheorien der Koordination lesen. Aktualisierte Beschreibungen für das

---

›Follow the actors« angesichts der Herausforderungen des Black-Boxing aktueller Software-Anwendungen tatsächlich realisierbar sind (Dieter u. a. 2019, 10).

<sup>36</sup> Dieser Ansatz ist gespeist aus der Akteur-Netzwerk-Theorie, in welcher Schüttpelz zuletzt stärker Momente des Sozialen hervorhob, wenn er die Bedeutung ethnographischer Beobachtung jener Abläufe und Vermittlungsschritte betonte, durch welche Medien im Modus ihrer »Verfertigung« konstituiert werden (ebd., 9f).

21. Jahrhundert, welche grundlegend aktuelle mediale Figurationen, Mechanisierungen und Automatisierungen, sowie Koordination der Warenproduktion als Medientheorie beschreiben, sind derweil rar. Daher konzentriert sich die vorliegende Studie genau auf diese Aspekte.

Drittens, ›kooperative Rahmen‹: Mit der Unterscheidung in kommunikative und kooperative Medien kann die Rahmenanalyse entsprechend modifiziert werden. Traditionell untersuchte die Rahmenanalyse, bezogen auf kommunikative Medien, Wörter, Bilder, Aussagen und Präsentationsweisen, die eine Aussage konstruieren sollen. Dieses Metier ist für Medien der Kooperation zu erweitern durch die Felder der automatisierten Registration und Adressierung, der Transaktionen und Delegation, des Prozessierens und der Koordination »von Körpern, Apparaten und Arbeitsteilung« (Gießmann 2018, 108). Somit orientieren ›kooperative‹ Rahmen das Zusammenwirken in nicht-humanen/humanen Figurationen, sodass Kooperation überhaupt einen Sinn ergibt. In Bezug auf Datenbanken bestünde ein solcher Rahmen im (Ein)-Verständnis über die Verwendung von ›Software‹ und einem Set spezifischer Fähigkeiten und kulturtechnologischer Vollzüge, um vernetzte Human-Computer-Interfaces zu nutzen. Häufig bleibt das Soziale dieses Vorganges unbeobachtet, indem das, was sich als gesellschaftlicher Verfertigungsprozess lesen lässt, als Technologie postuliert wird. Jedoch spielt das soziale Erlernen des Umgangs mit technologischen Artefakten eine wesentliche Rolle dafür, dass diese ihre Funktion überhaupt erfüllen können.

Ein anderer sozialer Rahmen in Bezug auf Datenbanken betrifft kollektive Vereinbarungen darüber, welche Topologien verwendet werden, welche Adressierungen, welche Kategorien, um eine Aussage zu treffen, beziehungsweise um eine Domäne zu beschreiben, in der Datenbanken überhaupt erst ›gültig‹ werden. Diese Rahmen überlagern einander. Bezogen auf Momente des individuellen Erkennens erlauben Rahmen den Individuen ein Verständnis der jeweiligen Situation und der eigenen Situiertheit darin (Goffman 1974, 39). Indem Nutzer bestimmte Rahmen anerkennen, zum Beispiel die Abfragesprache SQL, eröffnet sich ihnen überhaupt die Möglichkeit, einer Datenbank mit Abfragen zu begegnen. Aus den kooperativen Rahmen folgt das Konzept der ›kooperativen Figuration‹, welches die Komplexität von Datenbankpraktiken besser zu adressieren vermag, als ›System‹.

Viertens, ›delegierende Macht und gesellschaftliche Normative‹: Auch in kooperativen Figurationen sind Verteilungen von Macht zu beobachten. Das Interagieren verschiedener Akteure erzeugt Konflikte. Daher gilt es, das Soziale in der Analyse zu behalten: »Anstatt einen Ausschluss des Sozialen vorzunehmen, wie er sich in primär operativ angelegten Medientheorien immer wieder beobachten lässt (Siegert 2017:95–113), soll hier die Alltäglichkeit medialer Produktion, ihre Verkörperung durch die beteiligten Akteur\_innen ebenso wie die technisch-operative Realisierung als Grundlage gelten« (Gießmann 2018, 97). Daraus folge, dass Medien nicht bereits da seien, sondern immer wieder produziert werden müssten und zwar sozial.

Dies trifft in spezifischer Form auch auf delegierende, nicht-humane Akteure zu: auf Maschinen, Algorithmen, technische Netzwerke, Betriebssysteme, Datenbanken, die zwar agieren, jedoch ohne Bewusstsein. Die Macht der delegierenden, nicht-menschlichen Akteure ist wesentlich dadurch geprägt, dass sie gesellschaftliche Normative automatisiert fortschreiben. Denn Menschen glauben unhinterfragt an deren ›Objektivität‹. Die soziale Dimension ist daher gekennzeichnet nicht allein durch die Frage, wer spricht, sondern auch durch die Frage nach den Modi der automatisierten Fortschreibung von Normativen. Ein herausragender Kandidat diese Aspekte zu untersuchen, ist das Informationsmodell von Datenbanken, welches Einschlüsse und Ausschlüsse erzeugt.

*Zusammenfassend:* Figurationswandel, kooperative Rahmen, Medien der Kooperation, delegierende Macht und gesellschaftliche Normative sind jene Aspekte, welche mit der Figuration beobachtbar und

verhandelbar werden. Daher wird ›Figuration‹ anderen Konzepten, wie System, Netzwerk, Ökologie, Digitale Milieux oder Assemblage vorgezogen. Die sprachliche Nähe zur Konfiguration ist nicht unbeabsichtigt.

Die hier dargelegten Verknüpfungen zwischen Medienkulturtechnikforschung und Infrastructure-Studies stellen für die Softwarestudies vielfältige Beschreibungsebenen dar. Dem Unsichtbar-Werden von Datenbanken sind Beschreibungen des Registrierens/Identifizierens, Delegierens und Koordinierens entgegen zu setzen. Die Verwendung von ›Figuration‹ erlaubt es, das Konzept ›System‹ historisch zu verwenden, ansonsten aber eine aktualisierte Beschreibungsebene, welche die beständige Reproduktion kooperativer Medien adressiert, aufzurufen. Welche Potentiale und Problematiken diese Konzepte im Zugriff von Geschichtsschreibung entfalten, diskutiert der nächste Abschnitt.

## 1.3 Historiographie im Spannungsfeld des Ost-West-Diskurses

»Vom Standpunkt einer anderen Gegenwart aus, erkennen wir, wie in jedem Feld der Geschichte, in der Vergangenheit neue Dinge« (Haigh 2018, 4).<sup>xvi</sup>

### 1.3.1 Überlappende Zugänge

Um die Datenbank nicht länger als unhintergebares, technologisches Objekt und als scheinbar technologische Konstante zu definieren, ist neben dem kulturtheoretischen und infrastrukturellen Zugängen, eine genealogische Ebene einzuziehen, welche zum Teil auch diskursanalytisch vorgeht. Wie stehen diese Zugänge zueinander? Die Analyse von Datenbanken-Praktiken verfolgt eine Pendelbewegung zwischen Materialitäten und Praktiken, Diskursen und Genealogien. Materialitäten in Bezug auf die Datenbank sind gekennzeichnet durch die konkreten infrastrukturellen Formationen, welche sich um Fragen der Daten- und Symbolverarbeitung und -speicherung konzentrieren. Zu den Materialitäten zählen Hardwarekonfigurationen, vorrangig aber Softwarekonfigurationen, wie beispielsweise die Optimierungskomponente, Routinen zur Transaktionsabwicklung oder das Informationsmodell in Datenbankmanagementsystemen.

Im Umgang mit diesen Materialitäten und in deren Etablierung sind diskursive Praktiken beobachtbar, welche die Umgangsweisen legitimieren. Anhand von Zeitungsartikeln und Betriebsanleitungen wird den diskursiven Formationen nachgegangen, häufig in Bezug auf Institutionen, wie beispielsweise den zentralen DDR-Computerhersteller Robotron oder die Technische Universität Dresden: Welche Begründungszusammenhänge werden angegeben, wie wird das eigene Handeln legitimiert, welche Einschlüsse und Ausschlüsse lassen sich aufzeigen? Genealogien versprechen kein lineare und vollständige Geschichtsschreibung, vor allem keine Ursprungserzählung, wohl aber das Aufleuchten von Zusammenhängen über größere Zeiträume. Sie erlauben daher in größerem Maße als die klassische Geschichtsschreibung den Kontingenzen und zwangsläufigen Ungewissheiten und Leerstellen der Ereignisse gerecht zu werden.

Dabei fokussiert diese Studie auf die Felder der Verwaltung und des Managements, des Büro und der Organisation.<sup>37</sup> Die genealogische Trajektorien in die vorhergehenden Jahrhunderte und deren Bemühungen um Informationsverwaltung und -analyse mittels Karteikarten und Katalogen werden ebenso aufgerufen, wie die sich steigernden Informationsbedürfnisse der industriellen Produktion des Industriezeitalters. Mit dieser Setzung erschließt sich auch, warum der zweite Teil, im Unterschied zum dritten, historischen Teil, Fragen wie die relationale Operationalität, die Tabelle als Interface und die Transaktion als Prozess aufruft: Eine rein historische Perspektive würde die Dimensionen der Erstellung und des Gebrauchs relationaler Datenbanken zwar darstellen, nicht aber erklären können.

Das Erklären jedoch hat Parikka als zentrales Motiv der Medienarchäologie markiert: »alle archäologischen Ausgrabungen in die Vergangenheit zielen darauf ab, die aktuelle Situation zu erklären« (Parikka 2012, 6).<sup>xvii</sup> Die heutige Durchdringung des Alltags mit Datenpraktiken bliebe ohne ein genaueres

---

<sup>37</sup> Auch wenn das DDR-Innenministerium über eine um Überwachungsfunktionen erweitertes Personenmelderegister *Persönendatenbank der DDR (PDB)* verfügte und das Ministerium für Staatssicherheit für die Auslandsspionage die Recherche-Datenbank *System der Informationsrecherche der HVA (SIRA)* einsetzte, zeigt sich als Haupteinsatzfeld in der DDR die Datenverarbeitung in Produktion und Verwaltung.

Verständnis der Emergenzen bzw. Variantologien (Zielinski 2006, 7) von Datenbanktechnologien und -kulturen des Gebrauches hinter den Oberflächen der User-Interfaces verborgen.

### 1.3.2 Periodisierungen

In der Arbeit greifen drei Periodisierungen ineinander, welche sich jeweils auf die Micro-, Meso- und Macroebene der Untersuchung verteilen.

Erstens, knüpfen die hier untersuchten Genealogien an Periodisierungen an, welche von einer Unterteilung nach Jahrzehnten oder Krieg- und Eroberungen abweichen und im 16. Jahrhundert einsetzen. Sie sind markiert durch den langen Übergang vom Souverän, der personalisierten Herrschaft von Adel, König und Kirchenoberhäuptern und deren unmittelbar körperlichen Disziplinierungsmaßnahmen zu neuen Formen der Gouvernementalität und damit verbundenen Kulturtechniken der Bestrafung und Überwachung.<sup>38</sup> Diese waren gekennzeichnet durch eine Verinnerlichung ›richtigen‹ Verhaltens durch die Subjekte selbst, durch Normativität und Panoptismus. Diese Periode schuf wichtige Voraussetzungen dafür, auf welche Weise spätere Datenbanktechnologie auftrat. Denn sie fing an, das Individuum und die Gegenstände zu markieren, durch ein System von Adressen, Verzeichnissen und Ausweisen. Sie begann zudem die Verortung innerhalb der Masse zu explizieren, in Schule, Militär und Fabrik. Im vorliegenden Text werden Momente dieser Periode vor allem im Abschnitt 2.2 *Operationalisierung – Tabellieren* und 2.3. *Koordination – Transaktionen* aufgerufen.

Einsetzend mit den ›kybernetischen‹ 1950er Jahren wird die Disziplinargesellschaft um die Kontrollgesellschaft erweitert und transformiert. Dieses Konzept formulierte Gilles Deleuze 1990 in dem kurzen, einflussreichen Beitrag *Postscript on the Societies of Control*. Er koppelte es eng an zeitgenössische Medien und Computer, und markierte als eine der wichtigsten Folgen eine Stärkung des Individuellen: »Wir bemerken, dass wir es nicht mehr mit dem Paar Masse/Individuum zu tun haben. Die Individuen sind ›dividuell‹ geworden und Massen zu Stichproben, Daten, Märkten, oder ›Bänken‹« (Deleuze 1992, 5)<sup>xviii</sup>. Damit spricht er an, was wir heute mit Techniken wie Likes oder Self-Tracking in seiner vollen Durchsetzungsphase erleben: eine individualisierte Gouvernementalität, welche die Subjekte verführt, fortwährend Daten zu produzieren, die in den kapitalistischen Kreislauf eingeschleust werden.<sup>39</sup> Die Kapitel 2.1 *Formatisierung – Strukturierte Daten in der relationalen Algebra* und 3.2 *Robotron und SAP: Sozialistisches und kapitalistisches Enterprise Resource Management im Vergleich* sowie 3.3 *Kalifornien, Technische Universität Dresden und Robotron Dresden: Das Problemorientierte Seminar »Datenbanken« und DABA 1600* sind im Zeitrahmen des Übergangs zur Kontrollgesellschaft angesiedelt.

Mit dieser Markierung eröffnet sich der Raum für höchstspannende Neufassungen der verfügbaren Analysen: Wenn man die DDR als eine Staatsform der Disziplinargesellschaft kennzeichnen möchte, erscheint das Ende der DDR in neuem Licht. Es wäre dann nicht mehr allein als finanziellen und ideologischen Kollaps zu interpretieren, sondern als Unfähigkeit von der Disziplinargesellschaft auf eine

---

<sup>38</sup> Die Deleuzeschen Periodisierungen stehen im vorliegenden Text insofern unter Spannung, als dass sie üblicherweise Analysen westlicher, kapitalistischer Formationen überspannen. Lassen sie sich umstandslos auf nicht-westliche und nicht-kapitalistische Verortungen übertragen? Es würde zu weit führen, die Potenziale und Grenzen dieses Analysewerkzeugs für die DDR und den osteuropäischen Realsozialismus exakt bestimmen zu wollen, auch wenn dies eine lohnenswerte und ausstehende Aufgabe ist.

<sup>39</sup> Der Autor ist an anderer Stelle unter Bezug auf Manuela Otts *Es lebe die Dividuation! Zur Notwendigkeit anderer Denkkonzepte angesichts zeitgenössischer Teilhabepraktiken* (Ott 2015) und Gerald Raunigs *Dividuum. Maschinischer Kapitalismus und molekulare Revolution* (Raunig 2015) intensiver auf Fragen des Individuellen und der Datenbank eingegangen, siehe *Epistemic Harvest – The Electronic Database as Discourse and Means of Data Production* (Hunger 2018).

Kontrollgesellschaft (Deleuze) umzustellen. Geht man davon aus, dass die Kontrollgesellschaft jene Formation ist, welche im Abflauen des Industriekapitalismus der Koordination sozialer Kräfte am ehesten entspricht, so ist das Ende des osteuropäischen, sozialistischen Experiments eine nachholende Re-Integration in die Formation der Kontrollgesellschaft. Dieser These können die Fallbeispiele in Kapitel 3. *Technoemergenz im Ost/West-Vergleich* neues Material hinzufügen, um sie einer Überprüfung zu zuführen.

Zweitens, beleuchtet die Untersuchung auf der Mesebene, zwei Periodisierungen der Computertechnologie und -kultur, die sich einerseits in post-experimentelle, zentralisierte Mainframe-Technologie und andererseits in die dezentralisierende Durchdringung des Alltages durch Computerpraktiken unterteilen lassen. Diese Periodisierung ist für Ost und West leicht versetzt geltend zu machen. Nach den experimentellen Prototypen ab Mitte der 1940er Jahre prägten die 1950er und 1960er Jahre große, zentralisierte Mainframe-Computer, welche nur für das Bedienpersonal unmittelbar zugänglich waren. In den 1970ern hingegen setzte mit Timesharing ein Prozess der Individualisierung des Computings ein, der mit dem Personal Computer in den 1980er Jahren zur allgemeinen Durchdringung des Alltagslebens mit Computern, d.h. symbolverarbeitenden Medien-Maschinen, führte. Für diese ›individualisierende‹ Phase ist eine Ausdifferenzierung der computer-kulturellen Milieus zu beobachten, die sich nicht länger allein auf Universitäten, Verwaltungen und Großunternehmen sondern auch auf kleine und mittlere Unternehmen und in die Haushalte und das Feld des Privaten verdichteten (Ceruzzi 2003). Besonders im Abschnitt 2.1 *Formatisierung – Strukturierte Daten in der relationalen Algebra* wird dieser Übergang mit dem Einsetzen des Time-Sharings thematisiert, tritt aber auch in Abschnitt 3.3 *Kalifornien, Technische Universität Dresden und Robotron Dresden: Das Problemorientierte Seminar »Datenbanken« und DABA 1600 zu Tage*.

Drittens, ist für die DDR eine Macro-Periodisierung im Anschluss an den Technologiemanager Gerhard Merkel, der an den Entwicklungen unmittelbar und entscheidend beteiligt war, verfügbar. So gliedert sich die Entwicklung der Rechentechnik und Datenverarbeitung in der DDR in vier Phasen:

- 1948–1960: Pioniere verwirklichen ihre Ideen
- 1958–1971: Starke staatliche Förderung von EDV und Automatisierung
- 1971–1983: Stagnation in Entwicklung und Anwendung der Rechentechnik
- 1983–1989: Mikroprozessor und CAD/CAM/CIM-Euphorie (Merkel 2006, 50–53)

Die ersten drei Perioden überschneiden sich mit den Zeiten des zentralisierten Mainframes, während der vierte Zeitabschnitt auf die allgemeine Durchdringung des Alltagslebens verweist. Im Unterschied zum Westen fällt auf, dass in der vorliegenden Arbeit Personalcomputer eine untergeordnete Rolle spielen, unter anderem weil sie in nennenswerten Stückzahlen erst zwei Jahre vor Ende der DDR zur Verfügung standen.

### 1.3.3 Aus dem Material heraus...

Ein besonderes Anliegen der Studie ist es, jenen Datenbank-Dokumenten und Inskriptionen Sichtbarkeit zu verleihen, die üblicherweise in der Gegenwart von einer Veröffentlichung ausgeschlossen sind, da sie als Betriebsgeheimnisse gelten. Ein historischer Zugang ermöglicht somit das Freilegen tieferer



medialer Schichtungen der Medien der Koordination. Ziel dieser Vorgehensweise ist es, eine Variatologie einer Mediengeschichte der Datenbank zu entwerfen, indem die Datenbank als »Referenzsystem« (Siegert 2015, 5) zur Analyse bürokratischer Datenverarbeitung thematisiert wird.

Die Untersuchung ist, um nicht theoriegeleitet mit vorkonfigurierten Vorstellungen zu arbeiten, aus dem Material heraus entwickelt und durch den Gang der Materialuntersuchung gegliedert. Angesichts dieser Vorgehensweise ist eine Reihe von historiographischen Problemen zu verzeichnen, welche im Folgenden angesprochen werden.

Erstens, ›Fremd-Machen‹: Die Untersuchung bezieht sich auf die relationale Datenbank in West und in Ost. Indem die geographische und topologische Unterscheidung West–Ost aktiviert wird, kommt eine problematische Binarität des Kalten Krieges erneut zum Aufruf. Inwiefern ist diese Binarität haltbar? Bei genauerer Materialanalyse erscheint zumindest in Perspektive auf Datenbanken diese Binarität nicht als Trennung, sondern als ein Kontinuum, welches sich zwischen den Parabel-Enden Ost und West aufspannt. Dies wird genauer zu zeigen sein.

Mit dem Historiker Ben Peters soll für den historischen Teil ein durch den sowjetischen Literaturkritiker Viktor Šklovskij geprägtes Konzept aufgerufen werden, das hilft, die Ost-West-Parabel neu zu formatieren: Ostranenie, deutsch in etwa Fremd-Machen, ist ein Verfahren, Bekanntes aus anderer Perspektive zu betrachten (Peters 2017, xiii). Dieses Verfahren ist nicht allein als literarisches, wie bei Šklovskij bekannt, sondern auch ein ethnographisches Verfahren, wie Latour anmerkte. Die historische Distanz einerseits und die kulturelle, politische und soziale Unterscheidbarkeit der Situation in der DDR ergeben eine spezifische Folie, welche es erlaubt, scheinbar Bekanntes und implizite Vorannahmen über ›die Datenbank‹ zu distanzieren und aus der Distanz die Differenz zu identifizieren.

Zweitens, ›Akteure oder Superheros‹: Eine partielle Fokussierung auf Personen, wie beispielsweise Edgar F. Codd (2.1 *Formatisierung – Strukturierte Daten in der relationalen Algebra*), Rolf Gräßler (3.2.3.2 *Konzeption der SOPS*) oder Birgit Demuth (3.3.4 *Forschungsergebnisse aus der Sektion Informationsverarbeitung*) ließ sich nicht gänzlich vermeiden. Diese Personen erwiesen sich als zentral für ein spezifisches Material, durch ihre Urheberschaft von Fachartikeln, oder einer Dissertation. Um zu vermeiden, dass damit ›Superheros‹ (vgl. Haigh/Priestley 2015) der Geschichte des Computing konstruiert werden, betont der Text die institutionellen Kontexte der Personen. In diesem Sinne folgen wir den Akteuren, um herauszuarbeiten, in welchen Kontexten sich Ideen und Konzepte entwickeln und fortpflanzen (Latour 1987, 21).

Drittens, ›entlarvte Teleologien‹: Die DDR zeichnete sich durch ein Spezifikum in der Technologieerzählung aus: Für die USA, stellvertretend für den Westen, ließe sich eine teleologische Erfolgsgeschichte erzählen, die eine zunehmende Durchsetzung relationaler Technologie als Siegeszug inszenieren würde. Sie würde sich einordnen in die Erfolgserzählung der Miniaturisierung der Rechentechnik, der zunehmenden Rechenleistungen und Speicherkapazitäten und der sukzessiven Durchdringung des Alltags mit komplexer Computingtechnologie. Allerdings hält die Geschichte zahlreiche Seitenpfade bereit, nicht-realisierte Potentiale, welche ein Schlaglicht darauf geben, dass die beschriebene Durchdringung alles andere als geradlinig, logisch aufeinanderfolgend oder systematisch erfolgte.

Die DDR liefert hierfür ein doppelt spannendes Argument: Sie verstand sich bis in die Mitte der 1970er Jahre selbst als Siegerin der Geschichte. Führung und Bürger erwarteten, dass der aus den Wissenschaften heraus begründbare technologische Fortschritt mit dem erwarteten Fortschritt einer sozialistischen Gesellschaft (so zumindest die vulgärmarxistische Interpretation durch den Marxismus-

Leninismus) in eins fallen würde. Genau vor diesem Problem – nicht der Selbstwahrnehmung eines Landes, sondern in der historischen Darstellung warnt Thomas Haigh: »Der tiefgreifende Fortschritt von Computer- und Kommunikationstechnologie wird genutzt, um Behauptungen zu legitimieren, dass die Gesellschaft in einem beispiellosen Tempo transformiert wird, obgleich technischer Wandel und das Wachstum der Produktivität in den vergangenen Jahrzehnten [im Westen, F.H.] heute schwächer sind, als zu Zeiten der industriellen Revolution« (Haigh 2018, 11).<sup>xix</sup> In der DDR kippte, wie in Teil 2 zu sehen sein wird, die teleologische Stimmung mit den zunehmend sichtbaren Problemen des Realsozialismus. Damit klafft ein spannender Spalt zwischen offizieller Ideologie und Alltagswahrnehmung der DDR-Bürger auf, der für zahlreiche Diskussionen bereits während der DDR führte. Dieser Spalt, diese Spannung, kann mitgeführt werden, wenn es darum geht, vergangene und gegenwärtige technologische Entwicklungen des Osten und des Westens zu bewerten: Inwiefern ist den verfügbaren Erzählungen zu trauen? Welche blinden Flecken gibt es, die den ideologischen Ost-West-Konstellationen entspringen?

Viertens, »Defizitgeschichtsschreibung vermeiden«: Viele der interviewten Zeitzeugen betonen, teils als entlastender Begründungszusammenhang, teils als Klage, den Mangel an moderner Rechentechnik in der DDR. Diese Wahrnehmung, das Ziehen von Vergleichen mit dem Westen, reiht sich in eine Situation ein, die als Mangelwirtschaft identifiziert wurde. Der ungarischen Wirtschaftswissenschaftler János Kornai, hatte das Phänomen für die sozialistischen Staaten der 1960er und 1970er Jahre untersucht und im Buch *Warenmangel als ein fundamentales Problem der sozialistischen Planwirtschaften und die ungarische Reform* (1983) veröffentlicht. Demnach zeichnet sich eine sozialistische Mangelwirtschaft durch einen Überschuss an Kaufkraft, verursacht durch Subventionierung und steigende Gehälter und einen Fehlen an einem entsprechenden Warenangebot aus.

Der Historiker Ben Peters verweist mit Kornai zudem auf die Probleme des »Vertikalen Aushandelns«, welche im Zuge der Fünfjahresplanung auf den verschiedenen Planungsebenen dazu führten, dass, um Spielräume aufrecht zu erhalten, unwahre Angaben über das Produktionsvolumen von Industriebetrieben gemacht wurden. Dies führte zu dem generellen organisatorischen Problem, dass alle Beteiligten wussten, dass sie zwangsläufig mit falschen Zahlen über Materialeingang und Warenausgang operierten und ihre Zahlen entsprechend änderten, wodurch die Pläne bereits bei Erscheinen unrealistisch waren (vgl. ausführlich Peters 2017, 72–77).

Es ist in der Darstellung daher eine Abwägung zu treffen, die die berechtigten Wahrnehmungen der Zeitzeugen dieser Problematiken ernst nimmt, ohne die techno-sozialen Entwicklungen in der DDR als Defizit gegenüber dem Westen zu beschreiben und damit als normatives Gefälle zu perpetuieren. Vielmehr verweisen diese Planungs- und Organisationsprobleme darauf, dass mit großer Wahrscheinlichkeit auch im Westen ähnliche Probleme des Planens und der Organisation zu verzeichnen sind, wenn auch auf anderer Ebene. Ein Beispiel dafür ist das teils spektakuläre, häufig äußerst teure, Scheitern der Installation neuer Enterprise Resource Planning Software, welches im Abschnitt 3.2.4 *Wall-dorf: SAP und die modulare Integration* angesprochen wird.

Fünftens, »kleine Länder«: Abgesehen von den Problemen der sozialistische Mangelwirtschaft erhält die Problemlage eine Wendung, wenn man sich vor Augen führt, dass die DDR im Unterschied zur UdSSR und zur USA ein vergleichsweise »kleines Land« mit entsprechend lokalen Ressourcen war. Dies macht sie in ihrer Zeit eher mit Industrieländern wie zum Beispiel den Niederlanden oder Schweden und der dortigen Softwareproduktion vergleichbar. Ein Schlaglicht auf die Validität dieser Überlegung wirft die wenig bekannte Zusammenarbeit zwischen der Universität Uppsala und der Medizinischen Akademie

Dresden am relationalen Datenbankmanagementsystem MIMER in den 1980er Jahren – die einzige Kooperation der DDR mit einem westlichen Land in diesem technologischen Feld (vgl. Abschnitt 3.1.5 *Übersicht über Datenbanksysteme in der DDR*). Die „Kleine-Länder“-Situation verdeutlicht, welche Perspektiven es sich lohnt einzunehmen, wenn eine Computer- oder Softwaregeschichtsschreibung jenseits einer reinen USA-Zentrierung erfolgen soll.

Fremd-Machen, Akteure oder Superheroes, entlarvte Teleologien, Vermeidung von Defizitgeschichtsschreibung und kleine Länder – diese Kontexte der Historiographie begleiten den vorliegenden Text. Zu den jeweiligen historischen Konstellationen geben die Kapiteleinleitungen Auskunft. Mit den für die Softwarestudies fruchtbar gemachten Konzepten der Kulturtechnikforschung, der Infrastructure Studies, der Figuration und mit der Ost-West-Historiographie im Rücken, stellt sich nun die Frage, auf welche Weise die Untersuchung konkret durchgeführt wurde.

## 1.4 Archäologie des Immateriellen: Materialien und Methoden

Wie lassen sich scheinbar technologische Konstellationen als soziale Praxen neu lesen? Um die verschiedenen historischen Perioden und kultur-technischen Zugänge in den verschiedenen gerahmten Figurationen dicht beschreiben zu können, wurde eine Vielzahl von Quellen konsultiert. Der Durchgang durch die folgenden Materialien und Methoden beleuchtet die materielle Basis für eine praxeologische Lesweise.

Es wäre falsch anzunehmen, dass es sich bei den Archivalien und Publikationen um rein passive, heute verfügbare Zeugnisse ihrer Zeit handelte. In ihrer Zeit spielten all diese Elemente – Zeitungsartikel, Bedienungsanleitungen, Sitzungsprotokolle, Gesetzestexte – eine aktive Rolle darin, die Technokulturen ihrer Zeit diskursiv anzureichern. Um dem Umstand gerecht zu werden, erhalten sie in einigen Kapiteln verstärkt eine aktive Rolle: In Form von Zitaten werden die Quellen aufgerufen ›direkt‹ zu sprechen.

### 1.4.1 Zeitungsartikel & Bedienungsanleitungen

Ausschlaggebend für die Forschungsfrage waren Primär- und Sekundärliteratur seit dem 18. Jahrhundert, mit Schwerpunkt auf den 1970er bis 1990er Jahren. Als Primärmaterial wurden zahlreiche Artikel aus Fachzeitschriften ausgewertet. Westliche Fachzeitschriften, hervorzuheben sind die *Communications of the ACM* und die *IEEE Annals of the History of Computing*, standen elektronisch zur Verfügung. Projekte wie Archive.org und Bitsavers.org stellen darüber hinaus auch Publikumszeitschriften wie die einflussreiche *Datamation* online.

Die benötigte DDR-Literatur hingegen war nicht als PDF verfügbar und auch nicht verschlagwortet. Daher wurden die beiden wichtigsten Fachpublikationen für Computertechnologie der DDR, die Monatszeitschriften *Neue Technik im Büro* und Industrie-orientierte *rechentechnik/datenverarbeitung* in den Universitätsbibliotheken Leipzig und Dresden für die Jahrgänge 1973–1989 (dem Ende ihres Erscheinens) gesichtet. Im Zuge der Sichtung wurden 178 Artikel identifiziert, die folgende Stichpunkte betrafen: 1.) Datenbank, 2.) relational, 3.) Rationalisierung, 4.) Einheitliches System Elektronischer Rechner (ESER), 5.) Sachgebietsorientierte Programmiersysteme (SOPS), 6.) spezifische DDR-Datenbanken: BASTEI, DBS/R, DABA 1600, INTERBAS, MIMER, REDABAS und weitere.

In Folge der Sichtung konnten anhand des gescannten und verschlagworteten Materials einzelne Themenkomplexe aufgestellt werden und Prämissen für die weitere Erforschung aufgestellt werden. Somit ist nur ein Teil des verfügbaren Materials in den Textkorpus eingeflossen, da beispielsweise das Ausführen der für die DDR wichtigen Systeme DBS/R (ein netzwerkorientiertes DBMS) und AIDOS (ein dokumentenorientiertes DBMS) den verfügbaren Zeitzredit überzogen hätten. Diese Material harrt der weiteren Forschung. Da die untersuchten Fachzeitschriften hauptsächlich Belange des beruflichen Umgangs mit Computern betrafen, entfaltet sich darin eine spezifische berufliche DDR-Kultur des Umgangs mit Hard- und Software. Nicht allein die Berichte und Artikel waren dabei von Interesse, sondern auch ihr Verhältnis zu anderen Elementen, wie beispielsweise der Umschlaggestaltung, das Editorial, Berichte von Messen und Werbung, Leserreaktionen und Listen mit in der DDR verfügbaren Buchpublikationen und Software, sowie Anzeigen mit Technik aus zweiter Hand.

Aufgrund ihrer aktiven, pädagogischen Vermittlungsleistung kann geltend gemacht werden, dass Publikationen wie *Neue Technik im Büro* und *rechentechnik/datenverarbeitung* in der DDR nicht nur diskursbeteiligt waren, sondern diskursbestimmend, da sie neue Entwicklungen aufzeigten, Kritik öffentlich machten und wichtige Repräsentanten der Computing-Kultur in der DDR darstellten. Da die Herausgeberschaft zumindest von *rechentechnik/datenverarbeitung* eng an die Industrie gebunden war – so war der Robotron-Direktor immer auch Co-Herausgeber – bildete die Zeitung eine Schnittstelle, welche es erlaubte, die Bedürfnisse, Kritik und Fragen der Leserschaft an die Hersteller zurück zu vermitteln.<sup>40</sup> Das Wissen um die normativen Kräfte führt dazu, scheinbar technische Diskussionen genealogisch und sozialhistorisch zu lesen.

Besonders für den dritten Teil der Arbeit wurden außerdem die Bedienungsanleitungen der Robotron-Produkte gesichtet, die in der SLUB Dresden im Magazin vorliegen. Zum Teil konnten antiquarisch erworbene Betriebsanleitungen der frühen Systeme von SAP gesichtet werden. Diese liegen weder gesammelt bei SAP vor, noch sind sie in Bibliotheken vorhanden, da offensichtlich die Publikationen der 1980er und 1990er Jahre (für das SAP R/2) durch aktuellere Literatur ersetzt wurden (SAP R/3 und folgende). Insgesamt haben sich die Betriebsanleitungen aus Sicht der Software-Studies als äußerst wichtig erwiesen, da nur durch deren Studium begründete Schlüsse gezogen werden konnten. In der teils schwierigen Suche nach frühen Softwaredokumentationen deutet sich an, dass im derzeitigen Bibliothekswesen wenig Bewusstsein für deren Forschungswert für spätere Generationen besteht.

#### 1.4.2 Interviews

Die sieben ethnographischen und historiographischen Interviews mit Zeitzeugen hatten zwei Zielstellungen: Sie waren einerseits explorativ angelegt, als Oral History (Brennen 2013, 124–157), im Hinblick auf die persönliche Involviertheit der Befragten, die zum Teil auch Protagonisten in einzelnen Abschnitten sind. Dadurch konnten Zusammenhänge und Abfolgen hergestellt werden, die auf persönlichen Erfahrungen beruhen und im publizierten Wissen nicht enthalten oder rekonstruierbar waren. Zweitens dienten die Interviews der Verifizierung/Falsifizierung historischer Einordnungen und Leerstellen, welche sich aus der Literatur ergaben. Diese Fragen erfolgten meist im späteren Teil der Interviews. Es ging nicht darum, qualitative Daten zu gewinnen, sondern individuelle Erfahrungen und Einschätzungen exemplarisch ins Verhältnis zu historischen Trajektorien zu setzen. Alle Interviews

---

<sup>40</sup> Dies war allerdings nicht der einzige Rückkanal. Die bei Robotron angestellten Softwareingenieure, welche Kundenprojekte betreuten, sorgten ebenfalls für Informationsrückfluss.

waren davon geprägt, dass die Interviewten – Ingenieure, Mathematiker, Programmierer, Manager – sich erst von der Fachkompetenz des Interviewenden überzeugen mussten, bevor die Fragen tiefergehend erörtert werden konnten. Da es häufig um technische Fachtermini ging, mussten die Interviewten erst in Erfahrung bringen, ob das was sie berichteten, verständlich für einen Außenstehenden, den Medientheoretiker, sein würde.

Die Interviews mit starkem zeithistorischen Bezug waren einer kritischen Sichtung zu unterziehen, da sich darin Nostalgie und aktuelle Interpretationen der Vergangenheit überlagern konnten und Entscheidungen für einen produktiven Umgang mit diesen Schichtungen zu treffen waren (vgl. Angé/Berliner (Hrsg.) 2015, 2). Eine der zur Verfügung stehenden Möglichkeiten der ›Dechiffrierung‹ lag darin, die Aussagen doppelt zu lesen: Einerseits im Sinne von Historiographie und andererseits mit den Mitteln der Diskursanalyse als eine kulturelle Praxis, welche Vergangenheit und Gegenwart verbindet (ebd., 6). Historische Erzählungen wurden, soweit möglich, mit weiteren Quellen abgeglichen. Die Interviews wurden per Telefon oder bei persönlichen Treffen geführt und dauerten zwischen ein und drei Stunden. Sie wurden aufgezeichnet und vom Autor transkribiert. Anschließend erhielten die Interviewten die Transkriptionen zur Kenntnis mit der Möglichkeit zu Korrekturen und Ergänzungen.

Ein Großteil der Interviews erzeugte reichhaltiges und hochspannendes Material, dessen Umfang weit über den zur Wiedergabe verfügbaren Rahmen hinausgeht. Sie können und sollten daher für weitere Forschungsarbeiten der Software Studies und der DDR-Forschung oder zur Publikation in Betracht gezogen werden.

### 1.4.3 Archive

Ein Firmenarchiv des VEB Kombinat Robotron besteht nicht mehr. In einer Email wurde von ehemaligen Mitarbeitern mitgeteilt, dass dieses 1990 am Hauptsitz von Robotron (Grunauer Str. 2) in Container entsorgt wurde. Eine der Nachfolgefirmen, die Robotron Datenbank-Software GmbH Dresden baut derzeit eine museale Ausstellung sowie eine Sammlung auf, die ein Firmenarchiv allerdings nicht ersetzen kann.

Das Rechenwerk Computer- & Technikmuseum Halle ist eine Sammlung von 2000 historischen Computern und Rechenmaschinen mit dem Schwerpunkt der DDR. Darunter befindet sich auch der in 3.3 *Kalifornien, Technische Universität Dresden und Robotron Dresden: Das Problemorientierte Seminar »Datenbanken« und DABA 1600* angesprochene Robotron K 1600. Allerdings waren keine Installationsdateien der darauf laufenden Datenbank DABA 1600 auffindbar, sodass keine Möglichkeit bestand, diese im Betrieb zu untersuchen. Für die in 3.2 *Robotron und SAP: Sozialistisches und kapitalistisches Enterprise Resource Management im Vergleich* angesprochenen SOPS existiert zwar noch einzelne Hardware, aber die noch erhaltenen ECER-Systeme in Halle und im Heinz Nixdorf MuseumsForum in Paderborn sind nicht mehr betriebsfähig. Die Geräte lassen sich zwar anschalten, aber es ist nach derzeitigem Sichtsungsstand keine lauffähige Datenbanksoftware aus den 1970er und 1980er Jahren überliefert.

Im Bundesarchiv Berlin lagen größtenteils digitalisiert Gesetzestexte der DDR vor, sowie Beschlüsse des ZK der SED, des Politbüros und des Ministerrats der DDR, die sich auf die Gründung und strategische Ausrichtung des Kombinates Robotron bezogen. Dazu zählen auch Anordnungen unterhalb des Gesetzesebene. Diese sind größtenteils digitalisiert und online verfügbar.

Das Sächsische Staatsarchiv verfügt über allgemeine, teils disparate Unterlagen zum VEB Kombinat Robotron (Bestand 11594), die nur wenige relevante Treffer für das konkrete Vorhaben lieferten.

Ebenso vorhanden sind teilweise Unterlagen der für Robotron zuständigen SED-Grundorganisationen, letztere ist nicht digital erschlossen und nach 1981 unvollständig.

Das Universitätsarchiv der TU Dresden besitzt umfangreiche Unterlagen zu den Forschungsaufgaben und zur Forschungsplanung seit den 1970er Jahren in der Sektion 08 – Informationsverarbeitung, in welcher auch die Datenbankenforschung angesiedelt war. Dadurch konnten in 3.3 *Kalifornien, Technische Universität Dresden und Robotron Dresden: Das Problemorientierte Seminar »Datenbanken« und DABA 1600* historische Fakten komplettiert und verifiziert werden. Insgesamt zeigt das Archiv großes Potential für die weitere Erforschung der universitären Informatik in der DDR und der Forschung und Entwicklung in Dresden mit dem Kombinat Robotron, aber auch DDR-weit mit anderen Betrieben. Das ZEISS-Archiv ist einer der wenigen glücklichen Fälle, in denen Betriebsgeschichte des deutschen Reiches, des Dritten Reiches, der DDR und der Bundesrepublik Deutschland prinzipiell zugänglich ist. Ursprünglich Archiv des Kombinates VEB Carl Zeiss Jena, ist es 1990 übergegangen in das Archiv der Carl Zeiss Jena GmbH und stellt Fotografien, Druckschriften, Akten, sowie eine Gerätesammlung für Forschungsarbeiten zur Verfügung.

Durch das IBM-Archiv Poughkeepsie, NY wurden einzelne Dokumente, insbesondere Fotografien zu E.F. Codd zur Verfügung gestellt – eine Vor-Ort-Suche hätte möglicherweise weiteres Material zu Tage gefördert, allerdings schien das Archiv zum Zeitpunkt der Anfrage überlastet, da es stark nachgefragt ist.

Ein Firmenarchiv der SAP existiert bisher nicht. Seit 2019 ist das firmeninterne Team History Communications mit dem Aufbau eines Archives beschäftigt. Auf Anfrage wurden dem Autor freundlicherweise bisher unzugängliches Material zu den frühen SAP-Systemen zu Verfügung gestellt.

Der kombinierte Zugang zu Zeitungsartikel & Bedienungsanleitungen, Archiven und die eigens geführten, das Material kontextualisierenden Interviews ermöglicht einen praxeologischen Blick auf scheinbar technologische Figurationen, die als soziale Praxen neu gelesen werden können.

## 1.5 Navigation

Die verschiedenen Kapitel spielen an einer Reihe von Orten. Ein einführender Abschnitt führt die generelle Landschaft und die Werkzeuge ihrer Erkundung und Beschreibung auf. Teil I ist verortet im Normativ, dem Westen, zu dem die kapitalistischen Staaten Nordamerikas und West-Europas zu zählen sind. Teil II ruft als Kontrast den Osten auf, der sich vom geteilten Deutschland und dem Staatsgebiet der DDR über die sozialistischen Staaten des RGW erstreckt. Allerdings wird die Darstellung hier mit westlichen Entwicklungen verflochten, um beide miteinander ins Verhältnis zu setzen. Diese Teilung korrespondiert mit den untersuchten zeitlichen Periodisierungen und löst sich auf in der abschließenden Zusammenfassung.

### Teil I

Das Kapitel 2.1 *Formatisierung – Strukturierte Daten in der relationalen Algebra* geht von der 1970 erschienenen zentralen Veröffentlichung aus, in welcher die relationale Logik der Datenorganisation postuliert wurde. Es zeichnet von dort aus Genealogien nach, indem jene Elemente aufgesucht werden, welche das Konzept formten, Daten als Relationen zu strukturieren. Dazu zählt die semantische Ablösung der Information von der maschinellen Struktur stapelverarbeitender, sequentieller Speicherformationen und die Verwendung eines Informationskonzepts, demnach Bedeutung durch Referenz

zugeschrieben wird. Voraussetzung dafür ist die Strukturierung von Information als mathematisch-algebraische Menge, deren Ein- und Ausschlüsse in einem Informationsmodell niedergelegt werden. In dem Kapitel wird somit eine erste historisch-genealogische Schicht des konzeptuellen Umgangs mit Daten in unterschiedlichen institutionellen Kontexten der 1950er bis 1970er Jahre in den USA frei gelegt, von der aus weitere Erkundungen zum Tabellieren und der Aufzeichnung von Transaktionen einzuholen sind.

Das folgende Kapitel 2.2 *Operationalisierung – Tabellieren* dient der tiefen Genealogisierung von Information als Formation mit Beispielen ab dem 17. Jahrhundert. Es untersucht die Materialität der Tabelle und wie diese entlang ihrer Constructio gelesen und durch Ordnen, Sortieren, Adressieren, Filtern und das Beseitigen von Leerstellen operationalisiert wird. Die bereits angedeutete Auffassung von Information als Formation wird genealogisch grundiert und in diesem Zuge die Tabelle als Grundlage relationaler Datenbanken erstens im Registrieren/Identifizieren, zweitens im Delegieren und drittens in der Koordination verortet. Das Kapitel widmet sich weiterhin einer Reihe von Tabellenpraktiken, die sich aus dem jeweiligen Informationsbegehren ergeben und anhand historischer Beispiele das Herstellen von Wissensordnungen, statistische Erhebungen, das Durchführen von Transaktionen und das Erarbeiten mathematischer Tabellenwerke beschreiben. Es schlägt schließlich den Bogen von der Tabelle zur algebraischen Repräsentation von formatierten Informationsmengen in relationalen Datenbanken und zu der Frage, welche spezifisch neuen Möglichkeiten der Abfrage sich mit Hilfe der Structured Query Language (SQL) aus der Übertragung von der Tabelle zur relationalen Menge ergaben.

Im Kapitel 2.3. *Koordination – Transaktionen* wird zuerst anhand der doppelten Buchhaltung gezeigt, wie diese kaufmännische Praxis neue Formen der Kooperation und des Delegierens in der Frühneuzeit etabliert. Diese neuen Formen der Verzeichnung schaffen nicht nur eine neue epistemische Ebene, auf deren Grundlage kaufmännisches Wissen anhand von Analysedaten reorganisiert wird, sie ermöglicht auch zunehmend abstrahierte Kapitalverhältnisse im Frühkapitalismus. Dieses Wissen fließt schließlich aus den kaufmännischen Sphären zum Regenten, um dort als statistische Regierungstechnik zu reüssieren. Zweitens wird untersucht, wie die kooperative Kulturtechnik der Transaktion in Computern informatisiert wird. Am Beispiel von American Airlines wird der Wandel von Datenbanken als Informationssystem zu Datenbanken als Informations-Managementsystem analysiert, in dessen Gefolge On-Line Transaction Processing (OLTP) und On-Line Analytical Processing (OLAP) neue Technologien der Koordination nicht nur verkörpern, sondern auch ›vermaschinisieren‹.

## Teil II

Mit dem Kapitel 3. *Technoemergenz im Ost/West-Vergleich* beginnt der zweite Teil der Untersuchung.

Der Abschnitt 3.1 *Kontexte der Fallstudien* eröffnet den zweiten Hauptteil und führt als Vorspann eine Reihe von Kontexten ein, die vor allem jene adressieren, denen die DDR-Spezifika nicht bekannt sind. So werden überblicksmäßig Institutionen der Planung und Lenkung dargestellt, das Kombinat Robotron als Industriebetrieb des DDR-Computings eingeführt, das Wirtschafts-Embargo des Coordinating Committee on Multilateral Export Controls und seine Auswirkungen auf den DDR-Import von Computertechnologie diskutiert und die Aneignung von Software und deren Wiederveröffentlichung durch Robotron als Wissensaneignung kontextualisiert. Teil dieser Einführung, und in dieser Form auch für DDR-Kenner relevant, ist eine zeitliche und topologische Übersicht der überraschend vielfältigen Datenbanksoftware der DDR.

Zentral für den Vergleich industrieller Datenbank-Anwendungen in Ost und West in der Ära der Großrechner ist der Abschnitt 3.2 *Robotron und SAP: Sozialistisches und kapitalistisches Enterprise Resource Management im Vergleich*. Der Abschnitt ist ein grundlegender Beitrag zu den Genealogien kooperativer Medien. Er zeichnet nach, wie in Folge einer vereinheitlichten Dokumentation von Produktionsabläufen ab 1970 in der DDR eine modulare Software-Sammlung, den SOPS, für zahlreiche Anwendungsfälle industrieller Produktion und Koordination geschaffen wurde. Anhand von Fallbeispielen werden Aspekte der Implementierung und Nutzung in DDR-Großbetrieben diskutiert. Dem wird eine bisher nicht vorhandene Genealogie der frühen SAP-Softwareanwendungen gegenüber gestellt, die im Westdeutschland der 1970er ähnliche Aufgabenbereiche abdeckten. Beide, SOPS und SAP, beziehen sich auf verschiedene Weisen auf eine Reihe von IBM-Anwendungen der 1960er Jahre, die als Material Requirements Planning (MRP) bekannt waren und Ausgangspunkt heutiger Wirtschaftsoftware-Figurationen sind. In deren Zentrum sitzt das (relationale) Datenbankmanagementsystem, welches sich konzeptuell nach und nach herauschälte.

Der ersten relationalen Eigenentwicklung der DDR – DABA 1600 – widmet sich der Abschnitt 3.3 *Kalifornien, Technische Universität Dresden und Robotron Dresden: Das Problemorientierte Seminar »Datenbanken« und DABA 1600*. Die Darstellung der Genese von DABA 1600 als eigenständige Entwicklung beleuchtet epistemische Prozesse des Wissenstransfers über geographische und politische Systemgrenzen hinweg und verschärft die Frage nach den Unterschieden der politischer Systeme. Wieso gliederten sich die eingesetzten Medien und Denkwerkzeuge der Koordination und der Datenverarbeitung in dem weitreichenden Maße, dass das relationale Modell nicht als Technologie, sondern als Konzept, aus dem Westen in den Osten übertragen werden konnte? Die Darstellung der Entwicklung von DABA 1600 leistet zusätzlich etwas anderes: Sie zeigt die konkrete Entwicklung eines Mediums der Kooperation auf.



## 2. Genealogien relationaler Datenbanken

### 2.1 Formatisierung – Strukturierte Daten in der relationalen Algebra

»Einige von uns aus der Abteilung für Systementwicklung hatten versucht, sie zu lesen – wir verstanden nicht, wo vorn und hinten war. [Lachen] Zumindest damals schien es wie ein wirklich schlecht geschriebener Aufsatz: ein wenig Ausgangslage, und dann direkt in die Mathematik« (Irv Traiger, in: McJones 1997, 8).<sup>xx</sup>

»In dieser Studie geht es um die Anwendung elementarer Relationstheorie [...]« beginnt einer der folgenreichsten Aufsätze der jüngeren Informationstheorie, um dann zu präzisieren, worauf die Relationstheorie angewendet werden soll, und zwar auf »Banken formatierter Daten« (Codd 1970:377).<sup>xxi</sup> Er erschien 1970 in der Juni-Ausgabe der *Communications of the ACM* die als zentrale Publikation der Berufsorganisation Association for Computing Machinery diente.<sup>41</sup> Vorgänger dieser Veröffentlichung war die IBM-interne Forschungsschrift *Derivability, Redundancy, and Consistency of Relations stored in Large Data Banks*, welche auf den 19. August 1969 datierte, und am IBM Research Center in Yorktown Heights entstand. Eingangs wird in dieser Urversion das Programm vorgestellt:

»[Die relationale Algebra] stellt die Mittel zur Verfügung, um Daten allein durch ihre natürliche Struktur zu beschreiben: das heißt, ohne dass sie durch eine zusätzliche Struktur zum Zwecke der maschinellen Repräsentation überlagert werden. Somit bietet sie die Grundlage für eine höhere Abfragesprache, welche eine maximale Unabhängigkeit zwischen Programmen einerseits und maschineller Repräsentation andererseits leistet. Ein weiterer Vorteil der relationalen Sicht besteht darin, dass sie eine solide Grundlage für die Behandlung von Ableitbarkeit, Redundanz und Konsistenz der Relationen erschafft [...]« (Codd 1969, 1).<sup>xxii</sup>

In Aussicht wird gestellt, Daten unabhängig von Computerhardware, entlang einer »natürlichen Struktur« zu beschreiben, wobei sich die »Natürlichkeit« auf die Wahl der mathematisch-logischen Mittel bezog. Des weiteren soll diese Struktur mit Hilfe einer Abfragesprache durchsuchbar und bearbeitbar sein und zwar unabhängig von der umgebenden Struktur des Anwendungsprogrammes, in welches die Datenbank eingebettet ist. Schließlich zielt das Konzept darauf ab, »Ableitbarkeit, Redundanz und Konsistenz« von miteinander verknüpften Informationen mit algebraischen Mitteln zu garantieren, damit diese Computer-automatisiert verarbeitet werden können.

Umrissen wird im Folgenden ein grundlegendes Programm der Informationsverarbeitung in Formationen: Woher stammen die Ideen? Welches Umfeld erlaubte deren Zusammenführung? Wie kann die relationale Algebra historisch verankert werden? Gibt es neben den medienhistorischen auch medientheoretische Anknüpfungspunkte? Welche Probleme versprach der Ansatz zu lösen?

---

<sup>41</sup> Die ACM ist neben den Publikationen der Berufsorganisation der Ingenieure *Institute of Electrical and Electronics Engineers* (IEEE) die wichtigste Plattform für die Publikation akademischer Texte über neueste technologische Entwicklungen im Feld der Computing Sciences in den USA.

Die Frage nach den ideengeschichtlichen Herkünften des relationalen Datenmodells ist bisher ungeklärt. Auch in dem 1990 herausgegebenen Kompendium, welches das relationale Konzept detailliert beschreibt, gibt der Autor keine Hinweise auf dessen Genese. Im Gegenteil, der Mathematiker und Informatiker Codd schreibt einleitend: »Es ist auch das erste Buch über das relationale Modell vom Erfinder dieses Modells. Alle die Ideen, die in diesem Buch in Bezug auf das relationale Modell geschrieben werden, stammen von mir, außer in jenen Fällen, in denen ich explizit andere Autoren erwähne« (Codd 1990, V).<sup>xxiii</sup> Es scheint, als sei die Idee der relationalen Algebra gleichsam aus dem Nichts entsprungen – ein Geniestreich.

Wenn die Bedeutung dieses damals neuen Ansatzes für die Informationstheorie herausgearbeitet werden soll, kommt die vorliegende Studie um eine akribische Rekonstruktion der Ausgangsstränge nicht umhin. Systematisch werden die in Codd 1970 angegebenen Quellen daraufhin untersucht, inwiefern sie in seine Überlegungen einfließen. Ganz wesentlich betrifft dies die Studien *A Computer System for Inference Execution and Data Retrieval* von Roger E. Levien und Melvin E. Maron (1967), *Feasibility of a Set-Theoretic Data Structure – A General Structure Based on a Reconstituted Definition of Relations* von David A. Childs (1968), und das Logik-Lehrbuch *An Introduction to Mathematical Logic I.* von Alonzo Church (1956). Zudem wird der Rolle Charles T. Davies' nachgegangen, dem Codd am Ende seines Aufsatzes dafür dankt, ihn von der »Notwendigkeit der Datenunabhängigkeit zukünftiger Informationssysteme« überzeugt zu haben.<sup>42</sup> Ziel ist es herauszuschälen, was die relationale Algebra zum derzeit erfolgreichsten und grundlegendsten Modell der Registrierung von Informationen, der Delegation zwischen menschlichen und nicht-menschlichen Akteuren und der logistischen Koordination von Prozessen (vgl. Schüttpelz 2016; Gießmann 2018, 108) gemacht hat. Im Durchgang durch die eben genannten Texte wird nicht nur der Emergenz des relationalen Konzeptes aus historischer Perspektive nachgespürt. Es wird auch eine Fundierung für eine medientheoretische Interpretation des Informationsbegriffes als Formation in der Figuration der relationalen Datenbank formuliert.

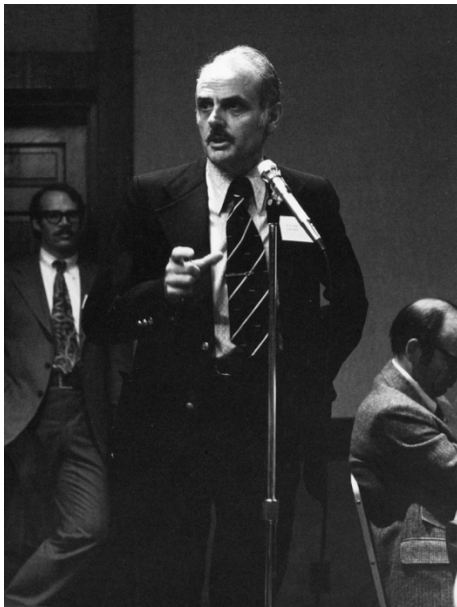


Abb. 2: Edgar F. Codd auf dem ACM SIGFIDET Workshop 1974 (Foto: Ben Shneiderman, IBM Archives Poughkeepsie).

---

<sup>42</sup> Mehrere Quellenangaben in *A relational model of data for large shared data banks* (Codd 1970) werden übergangen, da sich diese entweder auf Überblicksdarstellungen zu informatorischen Problemen seiner Zeit oder auf Betriebsanleitungen für vorhandene Datenbanksysteme, wie beispielsweise IBMs Information Management System (IMS), beziehen. Codd verwendet letztere, um in Abgrenzung zum hierarchischen Netzwerkmodell die Relevanz seines Ansatzes aufzuzeigen – ihre Ausführung soll hier aus Platzgründen unterbleiben.

## 2.1.1 Die Nutzung ins Zentrum rücken

»Dateimanagement-Systeme wurden für Magnetband-Speicher entwickelt [...]. DBMS wurden für Festplattenspeicher entwickelt« (Haigh 2009, 13).<sup>xxiv</sup>

### 2.1.1.1 Information Retrieval

Codds Veröffentlichung zur relationalen Algebra erschien in den *Communications of the ACM* nicht, wie man vielleicht erwarten könnte, unter dem Sachgebiet ›Datenbanken‹, sondern im Kapitel ›Information Retrieval‹. Mit dieser Verortung eröffnet sich eine Lesweise, die das relationale Konzept nicht als abrupten Beginn einer neuartigen Technologie verortet, sondern als Teil einer längeren Entwicklung – der Idee des Information Retrieval (Grier 2012).

›Information-Retrieval‹ (dt. etwa: Informationsabruf oder -rückgewinnung) entstammt den Anforderungen an besonders große Datenbestände, in denen die Nutzer ihre Dokumente verwalten und daraus Informationen ziehen. Die US Air-Force förderte durch die RAND Corporation entsprechende Studien. RAND, gegründet im Angesicht des Kalten Krieges 1948, widmete sich im Zuge des Sputnik-Schocks unter anderem auch der maschinellen Übersetzung, vor allem aus dem Russischen.<sup>43</sup> Jenseits theoretischer Konzepte scheiterten allerdings praktische Umsetzungen. Das Programm erwies sich als zu optimistisch, was die Potentiale maschinellen Übersetzens und künstlicher Intelligenz anging.<sup>44</sup> Eine Reihe offener Probleme führte Anfang der 1960er Jahre zur Neuausrichtung, und man wandte sich verstärkt Grundsatzproblemen der Linguistik zu. In der Folge produzierten Forschergruppen im Auftrag von RAND bis in die späten 1960er über 140 Aufsätze aus den Bereichen der Computational Linguistik, Computational Techniques, automatisierter Inhaltsanalyse, Information Retrieval und Psycholinguistik (Ware 2008, 143f.).

Im Teilprojekt *Cybernetic Research Data Project* forschten Roger Eli Levien und Melvin Earl Maron ab 1963 an einer Information-Retrieval Software. Ihr Interesse galt Computern als »Logik-Assistenten für die Analyse großer Datensammlungen von Fakten« (Levien/Maron 1967, 715).<sup>xxv</sup> Als Lösungsansatz schlugen sie das Prinzip des *Relational Data File* vor. Sie implementierten es auf dem Großrechner IBM 7044, einem Nachfolgemodell des wissenschaftlichen Rechners IBM 7090, das vorrangig an

---

<sup>43</sup> Als Sputnik-Schock wurde der Umstand bezeichnet, dass die USA vom Start des ersten sowjetischen Satelliten am 4. Oktober 1957 überrascht wurden und keine gleichwertige Technologie unmittelbar zur Verfügung stand. Den fehlenden Informationen über die sowjetischen Absichten begegnete man fortan mit der gezielten Übersetzung und Re-Emission öffentlich zugänglicher russischsprachiger Publikationen, die den Stand der Forschung in der Sowjetunion abbildeten, bis dato jedoch nicht auf Englisch verfügbar waren. RAND Corp. diente in seinen Ursprüngen dazu, sich ins Bilde zu setzen. Der Sputnik-Schock drückt auf einer zweiten Ebene die Reaktion der amerikanischen Öffentlichkeit aus, als klar wurde, dass die Sowjetunion in der Lage war, amerikanisches Territorium mit interkontinentalen Raketen zu erreichen. Der Sputnik-Satellit konnte auf der Trägerrakete im Prinzip durch einen atomaren Sprengkopf ersetzt werden. Bis dahin kamen zu diesem Zwecke auf beiden Seiten des Eisernen Vorhangs Langstreckenbomber zum Einsatz.

<sup>44</sup> So schlug der Philosoph und Linguist Yehoshua Bar-Hillel bereits 1957 in einer RAND-Studie über Informations-Retrieval-Systeme eine spezifische Logik der Datenorganisation vor, die der absehbar wachsenden Informationsmenge gerecht würde. Bar-Hillel mit dem spezifischen Blick des Logikers machte deutlich, dass die Fokussierung der Ingenieure seiner Zeit auf die maschinellen Grundlagen und die daraus abgeleiteten Konzepte zur Informationsverarbeitung »äußerst vage und wenig überzeugend« seien. Sein Schluss: Die Organisation der Informationsmengen müsse auf einer Mengen-Logik und Relationen basieren. Bar-Hillel erkannte jedoch auch, dass seine Vorschläge mit der vorhandenen sequentiellen Speichertechnologie nicht umsetzbar waren. Er formuliert damit eine Leerstelle der Informationsverarbeitung, die in den nächsten Jahren offen blieb, denn die Ingenieure seiner Zeit arbeiteten sich an den an Lochkarten und Magnetband orientierten index-sequentuellen Dateisystemen ab.

Universitäten und Forschungseinrichtungen eingesetzt wurde. Ihr daraus resultierender Aufsatz *A computer system for inference execution and data retrieval* (1965) war eine von zehn Quellenangaben: »[...] Die grundsätzliche Anwendung von Relationen auf Datensysteme [bezog sich größtenteils] auf deduktive Frage-Antwort Systeme. Levien und Maron bieten zahlreiche Verweise auf Arbeiten aus diesem Gebiet« (Codd 1970, 377).<sup>xxvi</sup> Die Studie selbst verfolgte einen induktiven Ansatz. Levien/Maron wurden zitiert, um das Feld, in dem sich der Autor bewegte, abzugrenzen. In der folgenden Diskussion von Levien/Maron sollen jene Momente beleuchtet werden, die unabhängig davon, ob induktiv oder deduktiv gearbeitet wird, für die relationale Argumentation eine Rolle spielten.<sup>45</sup>

Levien/Maron beziehen sich in ihrer Studie direkt auf das Konzept des Memex (Memory Expander) die Vannevar Bush 1945 populär formulierte. Memex war die Idee eines Arbeitsplatzes mit Bildschirmen, der das maschinenunterstützte Speichern, Auffinden und Verknüpfen von Informationen ermöglichen würde. Bushs Text gilt als einer der Ausgangspunkte der medialen Konfiguration, die wir heute als Computer bezeichnen, und für die Idee des Information Retrieval. Die Bezugnahme von Levien/Maron auf diesen visionären Aufsatz, lässt auch die Veröffentlichung zum relationalen Konzept in einem spezifischerem Licht erscheinen. Daher soll im Folgenden das Ineinandergreifen der Positionen anhand eines Durchgangs durch die drei Texte Bushs (1945), Levien/Marons (1965) und Cods (1970) untersucht werden.

#### 2.1.1.2 Nicht-Sequentieller Speicherzugriff

Einleitend formulieren Levien/Maron als Anforderung an eine Information-Retrieval Software, dass der Zugriff auf Daten der Logik menschlichen Denkens folgen solle. Der Zugang zu den Datenbeständen solle über die Inhalte erfolgen, und nicht über die technisch vorgegebene Speicherstruktur, die zum damaligen Zeitpunkt durch eine Logik des sequentiellen Zugriffs geprägt ist. Diese Erkenntnis ist grundlegend, denn in vielen frühen Softwarefigurationen gingen die Programmierer von der vorhandenen Hardwarestruktur aus und implementierten eine dementsprechende Logik. Levien/Maron wählten einen umgekehrten Ansatz: Erst die Logik und dann die Speicherstruktur. Sie identifizieren Direktzugriff-Speicher als Mittel ihrer Wahl. »Daher sollte die Datenbank 10.000 oder 100.000 Datenelemente enthalten, die in einem single-level Datenpool für den nicht-sequentuellen Zugriff auf der Basis des Inhaltes vorgehalten werden« (Levien/Maron 1967:716).<sup>46</sup> <sup>xxvii</sup> Konkret bedeutet das, die Daten nicht wie üblich, auf sequentiell zu lesenden Lochkarten oder Magnetbändern abzulegen, sondern auf Random-Access Festplatten, die einen Direktzugriff erlauben.

---

<sup>45</sup> Melvin E. Maron (1924–2016) war nach seinem Philosophie-Ph.D. 1951 an der UCLA zur IBM Research in San Jose gewechselt und arbeitete dort u.a. an Logik-Schaltkreisen. Danach wechselte er zur im Hochtechnologiebereich (Militär, Raumfahrt, Rechentechnik) tätigen Ramo-Wooldridge Corp., wo er bis 1958 an einem Informationssystem mitarbeitete. Dort entwickelte er die Idee des gewichteten Index.

Gewichtete Indexierung ist heute Grundlage vieler Suchmaschinentechnologien, sei es als Teil des PageRank in der Google Suche oder beim Open-Source Projekt Vufind, das für Bibliothekssysteme verwendet wird. Vgl. *An Historical Note on the Origins of Probabilistic Indexing* (Maron 2008).

Roger E. Levien (\*1935) hatte seinen Ph.D. 1962 an der Harvard University in angewandter Mathematik mit einer Arbeit zu *Studies in the Theory of Computational Algorithms* erhalten. Er arbeitete als Festangestellter von 1960 bis 1974 bei RAND Corp. Von 1969 bis 1974 war er als Adjunct Professor der System Sciences an der UCLA tätig. Danach wurde er Direktor des International Institute for Applied Systems Analysis (IIASA) in Laxenburg, Österreich.

<sup>46</sup> Ihre Software bezog sich auf einen Großrechner IBM 7044 mit einer IBM 1301 Festplatte, eine Konfiguration, die bis zu 200.000 »Sätze in Relationaler Information« speichern konnte. Implementiert wurde das Relational Data File als Stapelverarbeitung ohne direkte Nutzerinteraktion. Die Autoren schätzten allerdings ein, dass gerade die logische Analyse von Datenbeständen ideal für eine Mensch-Maschine-Interaktion im Direktzugriff wäre (ebd. 718, 721).

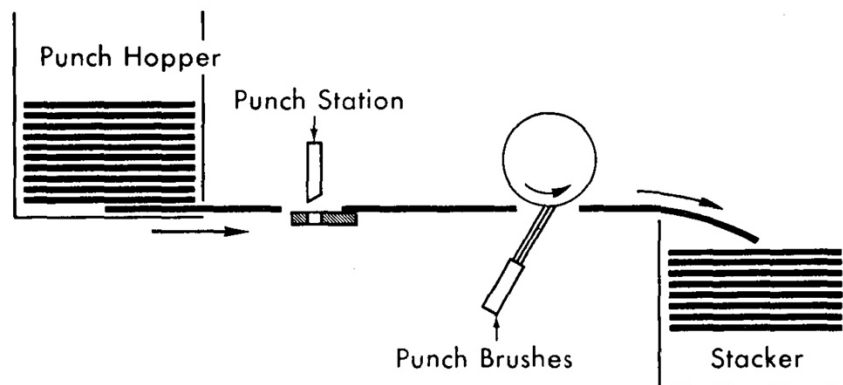


FIGURE 88. PUNCH FEED SCHEMATIC

Abb. 3: Lochkartenleser (Punch Brushes) und Stapel (Stacker). Jede Karte konnte 80 Zeichen speichern und das Gerät besaß einen Start- und Stoppknopf, um einen kompletten Stapel zu lochen. Die Illustration zeigt, wie die Daten sequentiell, wie am Fließband, an den Punch Brushes vorbeilaufen, welche die Spalten der Lochkarte zur Kontrolle abtasten, bzw. das Vervielfältigen von Kartenstapeln erlauben (IBM 1957, 127).

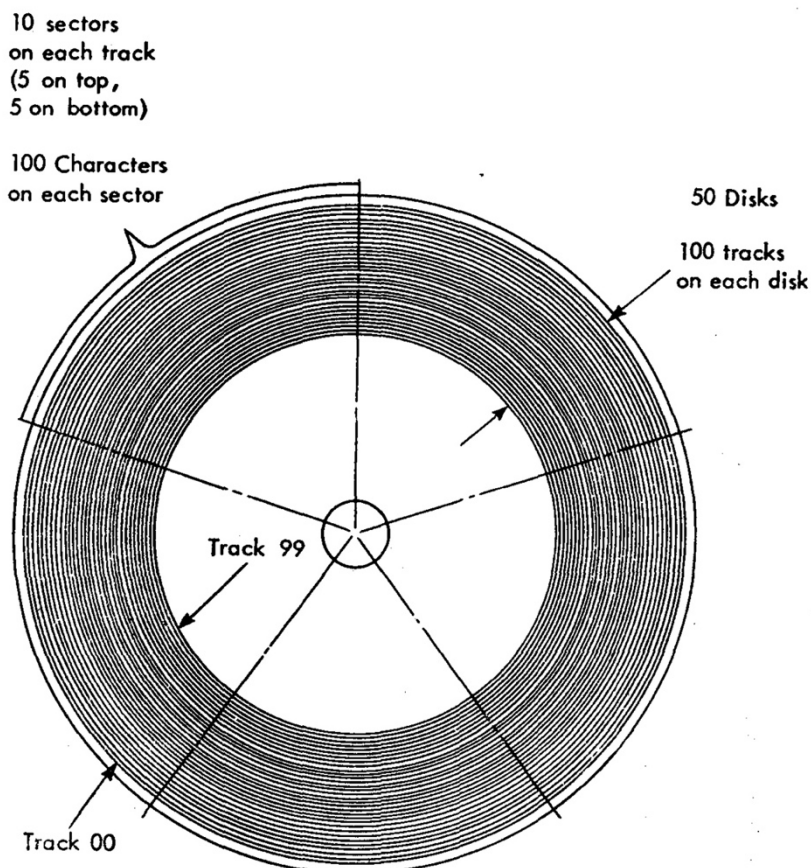


Abb. 4: Schematischer Aufbau der RAMAC Festplatte. »Die Disk-Spuren sind in Sektoren unterteilt. Es gibt zehn Sektoren pro Spur; fünf auf der Oberseite und fünf auf der Unterseite. Jeder Sektor kann 100 Zeichen einer Buchhaltungsdatensatzes speichern. Der Datensatz wird als eine Reihe von magnetischen Punkten im Sektor auf der Spur gespeichert. Indem zehn 100-Zeichen-Datensätze auf jeder der 100 Spuren und jeder der 50 Disks abgelegt werden, ist es möglich in diesem Gerät - 5.000.000 alphanumerische Zeichen zu speichern« (IBM 1957, 12).<sup>xxviii</sup>

Um zu illustrieren, was die sequentielle Verarbeitung von Bändern oder Stapeln programmierseitig bedeutete, können wir einen Blick in ein entsprechendes Beispielprogramm werfen, welches an die Sprache FORTRAN angelehnt ist und dazu dienen soll, alle Einwohner einer Straße in der Stadt ›Bradford‹ auszugeben.

```
1    PROGRAM readfile
2    INTEGER key, totalrecords, actualrecord
3    CHARACTER name, street, area, sex, job, filename
4    filename='personnel', totalrecords=9
5    OPEN (CHANNEL=1, FILE=filename, ACCESS='sequential', DEVICE='tape')
6    ! ----- Beginn Schleife -----
7    FOR actualrecord=1 TO totalrecords STEP 1
8        READ(CHANNEL=1)key, name, street, area, sex, job
9        IF(area='BRADFORD')
10           THEN DISPLAY name,street
11       ENDIF
12   NEXT actualrecord
13   ! ----- Ende Schleife -----
14   CLOSE(CHANNEL=1)
15   END
```

In den Zeilen 1 bis 3 wird der Programmname definiert und die Integer- und Textvariablen, die später verwendet werden sollen. Zeile 4 legt fest, welche Datei geöffnet werden soll und wie groß die Anzahl der darin auszulesenden Datensätze ist. Zeile 5 öffnet den Kanal, auf dem die Daten von Band in den Speicher übertragen werden. Die Schleife in Zeilen 7 bis 12 wird so oft durchlaufen, bis die Anzahl ›totalrecords‹ erreicht wird. Damit das Programm ›weiß‹, wo es sich gerade befindet, erhöht sich die Variable ›actualrecord‹ am Ende der Schleife jeweils um eins. Innerhalb der Schleife werden, egal ob benötigt oder nicht, alle Daten eines Datensatzes ausgelesen, damit der Lesekopf bis zum nächsten Datensatz vorrückt. Die eigentlich gesuchten Daten ›name‹ und ›street‹ werden, wenn die Variable ›area‹ den Wert ›Bradford‹ enthält, über eine IF THEN-Anweisung per DISPLAY ausgegeben. Der Rest der Daten bleibt im Speicher unbearbeitet liegen und wird mit Ende des Programms gelöscht, in Zeile 14 und 15.<sup>47</sup> Dieses Beispiel-Programm geht davon aus, dass die gesuchte Information direkt ab Bandanfang einzulesen ist. In der Realität waren jedoch mehrminütige Bandabwicklungen bis zur gesuchten Stelle zu absolvieren, bevor überhaupt Daten eingelesen werden konnten.

Während Loch- oder Magnetbänder ein beständiges zeitintensives Vor- und Zurückspulen und Lesen oder Schreiben eines kompletten Bandes oder Stapels erforderten (Abb. 3), erlaubte die Festplatte eine Verräumlichung des Datenzugriffs, der sich nunmehr in der Fläche organisierte. Waren bisher die Informationen sequentiell wie am Fließband auszulesen und zu verarbeiten, so erlaubt der wahlfreie Zugriff in der Fläche, alle Speicherstellen in konstanter Zeit zu erreichen (Abb. 4). Die Position im Speicher verliert an Bedeutung. Marcus Burkhardt bezeichnet diese Auflösung der Sequenz in die Fläche als eigentliche Revolution der neuen Speichertechnologie: »Nicht der Übergang vom Papierspeicher Lochkarte zum Magnetspeicher oder zu anderen elektronischen, aber weiterhin sequentiellen Speichertechnologien stellt die eigentliche Revolution in der Entwicklung neuer persistenter Speichermedien dar, sondern die Umstellung vom sequentiellen auf den direkten Speicherzugriff [...]« (Burkhardt 2015, 211).

---

<sup>47</sup> Beispiel angelehnt an (Yannakoudakis 1988, 9).

In der für die relationalen Algebra geschaffenen Abfragesprache SQL wird der Speicherzugriff, das Auslesen der Daten und die Übersetzung des abstrakten Datenbankschemas in eine physische Speicheradresse vom Datenbankmanagementsystem übernommen. In der Fläche der Festplatte verflüchtigt sich der zum Auslesen benötigte Zeitraum in den Bereich von Millisekunden. Der komplizierte und fehleranfällige FORTRAN Code, der von den Nutzern/Programmierern für jeden neuen Suchdurchlauf anzupassen war, stellt sich in SQL wie folgt dar:

```
SELECT name, street
FROM personnel
WHERE area = 'Bradford'
```

Die technischen Details des Zugriffes werden den Nutzern/Programmierern gegenüber opaque und irrelevant. Die Abläufe, die dazu führen, dass der Zugriff auf die Daten unter der richtigen Adresse erfolgt, steuert das unterliegende Datenbankmanagementsystem. Sie wurden von nun an in den computationalen ›Tiefen‹ der DBMS versteckt. Voraussetzung dafür war die Festplattentechnologie, die ab 1953 zur Verfügung stand, und die von Codd vertretene Auffassung, vom sequentiellen Datenzugriff Abstand zu nehmen, welche bei Levien/Maron, wenn nicht ihren Ursprung, so zumindest einen wichtigen Zeugen fand. Levien/Maron erkannten, dass sich ihre Informationssammlung nicht der linearen Speicherlogik des Bandes unterordnen sollte, sondern dass es anstrebenswert wäre, die logische Ordnung ihrer Information-Retrieval Software von der logischen Ordnung des Datenspeichers abzutrennen.

### 2.1.1.3 Abfragesprache mit Prädikatenlogik

Einige der Grundelemente der Prädikatenlogik, die wir später bei Codd und Chamberlin und spezifisch in der Abfragesprache für relationale Datenbanken *Structured Query Language* wiederfinden, wurden bereits von Levien/Maron hergeleitet: »Da wir es mit der logischen Analyse faktischer Daten zu tun haben, liegt es nahe, sich mit der formalen Logik auseinanderzusetzen als Struktur einer Informationssprache« (Levien/Maron 1967:717).<sup>xxix</sup> Ihr Vorschlag bezog sich auf eine deduktive Vorgehensweise. In deduktiven Datenbanken können Schlussfolgerungen, beispielsweise Zustimmung oder Ablehnung, formalisiert verarbeitet werden. Dazu werden Regeln aufgestellt, die mit Fakten kombinierbar sind – auf diesen Bereich bezogen sich Levien/Maron. Induktive Datenbankfigurationen dienen hingegen dazu, entweder einzelne Daten auszugeben oder Muster in Datenmengen zu erkennen. Codd übernahm von Levien/Maron die relationalen Ansätze der Datenorganisation, verabschiedete sich aber vom deduktiven Ansatz der Prädikatenlogik zugunsten eines induktiven Prädikatenkalküls, in dem Quantoren auf Mengen angewendet werden.

Am Beispiel der Verwendung von Schlüsseln lässt sich nachvollziehen, wie der Wissenstransfer zwischen beiden Ansätzen vonstatten ging. Levien/Maron schlugen aufgrund technischer Limitationen vor, maximal binäre, also zwei Prädikate zu verwenden. Größere Prädikatensammlungen sollten zu binären Prädikaten aufgelöst werden, wobei die aufgelösten Elemente über Schlüssel nachträglich wieder verknüpft werden könnten. Diese technologische Einschränkung auf binäre Prädikate ist 1969 nicht mehr nötig. Für das relationale Konzept liest es sich das dann so: »Eine übliche Anforderung an Elemente einer Relation ist es, andere Elemente der gleichen Relation oder Elemente einer anderen Relation zu vergleichen. Schlüssel sind ein Nutzer-orientiertes Mittel (aber nicht das einzige Mittel) solche Vergleiche auszudrücken« (Codd 1970, 380).<sup>xxx</sup> Hier wurde also auf die Verwendung von abstrakten

Schlüsseln abgezielt, meist sind das fortlaufende Zahlen, die aufeinander bezogen werden können. Levien/Maron formulieren abschließend: »Damit bringen wir eine relationale Informationssprache zur Anwendung« (Levien/Maron 1967:717).<sup>xxxii</sup> In ihrer deduktiven Software wurden Informationsabfragen nach dem Schema ›X received the Ph.D. degree from UCLA‹ möglich, wobei X die zu erfragenden Variable war.

Sichtbar wird hier, wie Codd an einem realisierten, praktischen Beispiel ein Verständnis dafür entwickeln konnte, dass Relationen einerseits durch Tupel darstellbar sind und andererseits die Tupel untereinander durch Schlüssel verknüpft werden können. Und es wird deutlich, dass Codd's Leistung auf einen breiteren Diskussionskontext aufbaute, denn Levien/Maron zählten sechs weitere Studien zur Entwicklung Algorithmen-basierter Aussagenlogik als Zeugen ihres eigenen Ansatzes auf.

#### 2.1.1.4 Formalisierte Dateneingabe

Der Medientheoretiker Markus Krajewski beschrieb die zunehmende Formalisierung und Diagrammatisierung von Informationen in der Sukzession des bibliographischen Zettels, der Karteikarte, des Formulars und der Lochkarte, »insofern die anfangs einfach linear auf dem Papier notierten Schriftzeichen immer stärker formatiert werden. Gleichzeitig unterliegt die Anordnung der Schrift auf dem Papier einer zweiten Formatierung, und zwar hinsichtlich der Position auf dem Blatt« (Krajewski 2007, 43). Damit weicht Krajewski's Begriff der Information als Formation ab von dem am Signal-Rauschen orientierten Informationsbegriff nach Shannon/Weaver. Aus diesem Blickwinkel ist in Levien/Marons' Studie bezeichnend, auf welche Weise die aufzunehmenden Informationen in die relationale Logik implementiert werden sollten. Auf den ersten Blick schienen sich Relationen der Informatisierung durch Encodierung zu verweigern: »Die Dateneingabe direkt in der relationalen Informationssprache könnte die Notwendigkeit späterer Übersetzungsschritte ausschließen; Allerdings wären weitreichende Anstrengungen nötig, um die große Menge von Sätzen, die zur Datenaufzeichnung notwendig sind, zu identifizieren, aufzuzeichnen und zu überprüfen; [...] Die Verwendung des Englischen zur Dateneingabe wäre für die Datenerheber komfortabel, doch für den Computer ungeeignet. Daraus folgt die Verwendung von Eingabefeldern« (Levien/Maron 1967, 718).<sup>xxxiii</sup> Warum Eingabefelder?

Mitte der 1960er Jahre, als Levien/Marons Text entstand, verfügten Computer noch nicht über Möglichkeiten zur graphischen Darstellung von Eingabefeldern. Der Zugriff erfolgte zeilenorientiert und textbasiert mittels schreibmaschinenähnlicher Terminals, die erst in den 1970ern und 1980ern zunehmend von Bildschirm-orientierten Zugriffsmöglichkeiten abgelöst wurden. Statt mit Eingabemasken am Bildschirm arbeitete man mit Papierformularen, deren Inhalte von Hilfskräften auf Lochband übertragen wurden. Dabei dienten die Formulare der Erzeugung disparater, maschinenverarbeitbarer Daten, wobei für jede Klasse von Daten (z.B. bibliographische, organisatorische, biographische) jeweils spezifische Formulare benötigt wurden. Der Übergang vom Papier-Formular zur Information als Formation ist zweifelsohne in vielen anderen frühen Datenerfassungsprojekten zu beobachten.

Hier ist die Aufnahme von Daten spezifisch dahingehend, dass sie einem Prädikatenkalkül unterworfen werden sollen. Mittels der eigens entwickelten Übersetzungssoftware FOREMEN, dt. etwa Vorarbeiter, wurden die Formulare in die gewünschte Prädikatenlogik übertragen und in die Datei binär eingeschrieben.<sup>48</sup> »Erstelle für jeden Autor *a* dessen Name in einem bibliographischen Formular für die Publikation *p* erscheint, den Satz *a* / AUTOR VON / *p* und füge ihn der Datei hinzu« (Levien/Maron 1967, 718).<sup>xxxiii</sup> So lautet in etwa eine Anweisung, die FOREMEN ausführte. Diese Vorgehensweise

---

<sup>48</sup> Ein Prädikat ist die Funktion einer oder mehrerer Variablen, deren Wert im Sinne einer Aussagenlogik ›Falsch‹ oder ›Richtig‹ sein kann. Eine Relation ist das Prädikat zweier Variablen.



dürfte als ein frühes Beispiel dienen, wie in mehreren Schritten eine informatorische Aussage durch die diagrammatische Organisation in einem Formular für eine Prädikatenlogik aufbereitet wird und einer relationalen Datenbank zur Verfügung gestellt werden kann.

#### 2.1.1.5 Nutzungszentrierung

Vannevar Bush beschrieb 1945 die Möglichkeiten der Memex aus einer Perspektive der Nutzung. Am Anfang stand die Fragestellung, wie vorhandene Informationsbestände rekombinierbar wären, und wie vorhandenes Wissen produktiv gemacht werden kann. Kann ich überhaupt wissen, über welches Wissen ich verfüge? Diese Perspektive der Verwendung, also eine Perspektive, die explizit nicht von den technischen Experten elektronischer Rechenmaschinen ausgeht, sondern von den Nutzern einer Informationsabfragesoftware, formulierten auch Levien/Maron. »Wie kann eine Sprache brauchbar für jene Nutzer entwickelt werden, die keine Spezialisten sind?«<sup>xxxiv</sup> (Levien/Maron 1967, 721). Die in dieser Frage formulierte Nutzerzentrierung, zweifellos aus Vannevar Bushs Memex-Ansatz stammend, unterscheidet Leviens/Marons und in der Folge auch Codds Argumentation signifikant von der des u.a. durch Charles Bachmann propagierten CODASYL-Netzwerkmodells. Levien/Marons Informationssystem sollte sich nicht an Programmierer wenden, sondern an Wissenschaftler und Manager, die in die Lage versetzt werden sollten, Informationen direkt abzufragen. Ähnlich argumentierte Codd in der Konfrontation mit Bachmann: »Ich verstehe nicht ganz, warum sich in den vergangenen sechs Jahren CODASYL nicht stärker auf andere Nutzer als die Programmierer konzentrierte« (Codd/Edgar F./Bachman 1975, 132).<sup>xxxv</sup> Während Codd den Umgang im direkten Dialog mit der Datenbank positionierte, sollten im CODASYL-Modell Programmierer durch die Datenmengen navigieren. Bushs' Memex sieht keinen zwischengeschalteten Programmierer vor. Die Nutzer sitzen an dem schreibtischartig organisierten Gerät und formulieren die Abfragen selbst (Bush 1945, Teil 6).

Allerdings sind Levien/Maron und Codds Veröffentlichungen selbst Zeugen von Veränderungsprozessen der Nutzerschaft, der Art und Weise der Nutzung und damit einhergehend des Begriffes Nutzer. An den Großcomputern und Zentralrechnern der 1950er und 1960er Jahre waren ›die Nutzer‹ spezialisierte Programmierer und Operateure, die den Abteilungen einer Firma oder akademischen Institution Informationsleistungen zur Verfügung stellten. Mit dem Aufkommen vergleichsweise kleinerer Computer ab Mitte der 1970er Jahre, die dezentral genutzt werden konnten, änderten sich die Nutzerschichten: Nutzer konnten nun auch qualifizierte Manager und Sekretärinnen sein, die teils vorgefertigte Programmpakete oder die von Programmierern vorkonfigurierten Softwarefigurationen verwendeten. Auf diese Weise rückte der Modus der Verwendung stärker an die jeweils gelebten managerialen und verwalterischen, mithin kooperativen Praktiken heran. Die Überlegungen schlossen diese ebenso ein, wie, z.B. eine Hausfrau, welche die besten Angebote des Supermarktes abfragen möchte (Codd/Date 1975, 95). Auf diese Weise näherte er sich aus der Perspektive der Datenhaltung dem Nutzerbegriff Vannevar Bushs an und dehnte ihn in Richtung alltäglicher bürokratischer, verwalterischer und managerialer Praxen aus.

#### 2.1.1.6 Kontext: Hyperlink oder Mengentheorie

Trotz der aufgezeigten Bezüge, gibt es keine Hinweise darauf, ob Codd den Artikel Bushs kannte. Hätte er ihn bei Gelegenheit gelesen, wäre er dabei in Abschnitt 7 auf folgende Stelle gestoßen:

»Für die assoziative Indexierung wird allerdings ein unmittelbarer Schritt benötigt; er basiert auf einer Einrichtung mit deren Hilfe jedes Objekt dazu gebracht werden kann, unmittelbar und automatisch ein anderes auszuwählen. [...]

Wenn der Nutzer einen Pfad anlegt, benennt er ihn, fügt den Namen in sein Code-Buch ein und gibt es per Tastatur ein. Vor ihm befinden sich die beiden zu vereinigenden Objekte, die auf benachbarte Sichtfenster projiziert werden. Unterhalb eines jeden befindet sich eine Anzahl von leeren Code-Feldern, und ein Zeiger ist so eingestellt, dass er für jedes Objekt auf eines davon verweist. Der Nutzer drückt eine einzelne Taste und die Objekte sind permanent verbunden. [...]

Des Weiteren können Objekte, die in großer Zahl zusammengefügt wurden, so dass sie einen Pfad ergeben, langsam oder schnell durchgesehen werden [...]. Es verhält sich genauso, als wären die physischen Objekte aus weit verzweigten Quellen gesammelt worden und zu einem neuen Buch gebunden. Es geht sogar darüber hinaus, denn jedes Objekt kann zu zahlreichen Pfaden verknüpft werden« (Bush 1945, Teil 7).<sup>xxxvi</sup>

Heute werden Vannevar Bush ›Pfade‹ mit dem Konzept des Hypertext-Links in Verbindung gebracht. Der Philosoph und Informatiker Ted Nelson arbeitete, inspiriert von Bush, seit den 1960ern an seinem Projekt Xanadu, einem der ersten Hypertext-Softwarefiguren. Dazu veröffentlichte er unter anderem 1965 ein Papier in den *Communications of the ACM*. Kann Codd die Xanadu-Interpretation gekannt haben? Wie ernst hätte er die Arbeit Ted Nelsons genommen?

Rufen wir uns in Erinnerung, dass 1970 das vom Hypertext strukturierte World-Wide-Web in seiner Durchsetzung mehr als zwei Jahrzehnte in der Zukunft lag, so besteht die Möglichkeit, dass aus damaliger Perspektive die Interpretation der Pfade als ›Hypertext-Links‹ nicht in Frage kam. Wer in den 1970er Jahren selbst eine Memex benötigte, konnte ein »Prozess, zwei Objekte miteinander zu verknüpfen« auch anders gelesen haben. Eine »assoziative Indexierung [...] mit deren Hilfe jedes Objekt dazu gebracht werden kann, unmittelbar und automatisch ein anderes auszuwählen« kann durchaus als relationale Algebra gelöst werden.

Bush's Vorschlag eines ›Pfad‹ würde zwar durchaus auf die Idee des ›Programmierers als Navigator‹ eines Charles Bachmann verweisen, der, im Rahmen des CODASYL-Komitees, das nicht-relationale Netzwerkmodell propagierte.<sup>49</sup> Jedoch löst sich mit Bushs Bemerkung: »Es geht sogar darüber hinaus, denn jedes Objekt kann zu *zahlreichen* Pfaden verknüpft werden« die strenge Sichtweise des Pfads als sequentiell zu rezipierende Abfolge auf. »Jedes Objekt kann verknüpft werden«, kann auch eine andere Bedeutung tragen: die Verknüpfung zweier Mengen von Informationsobjekten mittels Boolescher Algebra. Letztlich unterscheidet sich die Herangehensweise der relationalen Algebra von dem von Bush imaginierten ›Trail‹ dadurch, dass es nicht zu dauerhaften Verknüpfungen kommt, sondern die Assoziationen im Moment der Datenabfrage, ad hoc hergestellt werden.

#### 2.1.1.7 Zusammenfassung

Levien/Maron beendeten ihre Studie mit einer Reihe von offenen Problemen, die sich im Laufe des Projektes ergeben hatten. Codd entwickelte anhand seiner eigenen Fragestellung die aufgezeigten Ansätze weiter, jenseits deduktiver Frage-Antwort-Software. So sind Übereinstimmungen in den Punkten

---

<sup>49</sup> Die relationale Algebra ihrerseits ist in der Lage ein Netzwerkmodell abzubilden, umgekehrt aber nicht. Ein ›Trail‹ ist damit auch relational realisierbar.

Nutzerzentrierung, Information Retrieval, Nicht-Sequentieller Speicherzugriff, Information und Mengentheorie festzustellen.

Die Spur von Vannevar Bushs Memex verliert sich; sie wird in Codd 1970 nicht benannt. Als gemeinsamer Ausgangsort des Hyperlink-Projektes Xanadu, welches dem World Wide Web vorher geht, und des Relational Data File, welches für Codd's Überlegungen zur Relationalen Algebra produktiv wird, ist jedoch die 1945 imaginierte Medienmaschine Memex höchst spannend. Es ist eine Maschine, die mediale Inhalte speichert, indexiert und verknüpft. Jenseits des Computers als Rechenmaschine, als ›number cruncher‹, entwirft Vannevar Bush die Vision einer zeichenverarbeitenden und datenverknüpfenden Maschine. Diese wird nach und nach verwirklicht und zwar ab jenem Moment, ab dem die Entwicklung der Maschinenteknik eingeholt wird von der zunehmenden Bedeutung der Softwaretechnik. Und sie wird doch nicht verwirklicht, denn die Abfrage auf Knopfdruck, also ohne Query, existiert nach wie vor nicht.

## 2.1.2. Namen oder Bedeutung

### 2.1.2.1 Entitäten im Spiegel der Universalismusdebatte

Beim Erstellen und Befüllen von Datenbanken wird Wirklichkeit übersetzt. Sie wird in eine Struktur übertragen, die erlaubt, bestimmte Aspekte dieser Wirklichkeit maschinell zu verarbeiten. Jene Aspekte, die nicht durch elektronische Computer verarbeitbar sind oder deren Verarbeitung aus betriebsökonomischen Erwägungen nicht sinnvoll erscheinen, werden abgespalten.<sup>50</sup> Wie die Konzepte von ›Daten‹ im Verhältnis zu Formationen und zur Erzeugung von Bedeutung zu positionieren sind, erörtert der folgende Abschnitt.

Eine der geläufigsten Theorien der Informatisierung ist jene, die auf die Differenz von Signal und Rauschen nach Claude Elwood Shannon abzielt. Sie ist jedoch für das Verhältnis von Wirklichkeit und informatorischer Abbildung in elektronischen Datenbanken unzureichend, weil sie die mathematischen Grundlagen von elektromechanischen Schaltungen betrachtet, und desinteressiert ist, welchen Aussagegehalt das zu verarbeitende Signal hat. Fragen nach dem Aussagegehalt würden einhergehen mit Fragen danach, wie einem Gegenstand Bedeutung zugeschrieben wird – mit Fragen der Semantik (Shannon 1948).

Die Zuschreibung von Bedeutung findet in der relationalen Algebra in einem spezifischen Modus der Referenz und der Informations-Modellierung statt.<sup>51</sup> Diesem soll im Folgenden nachgegangen werden. In der Vorab-Publikation über die relationale Algebra, die 1969 intern bei IBM erschien, hinterließ

---

<sup>50</sup> Momentan wandelt sich dieses Verhältnis von Wirklichkeit und ihrer computertechnischen Verarbeitung. Unter dem Stichwort ›Big Data‹ sind eine Reihe von Technologien versammelt, die unstrukturierte Daten informatisieren können. Die Idee Big Data ist nicht allein ein technologisches, sondern auch ein ökonomisches Versprechen: Aus den gesammelten Daten seien potentiell Muster oder andere Informationen zu identifizieren, die eine zukünftige ökonomische Verwertung erlauben.

<sup>51</sup> Im folgenden Sprachgebrauch sind ›Informationsmodell‹ und ›relationales Modell‹ nicht miteinander zu verwechseln. Das ›Informationsmodell‹ bezieht sich auf ein logisches Modell von Ein- und Ausschlüssen, nachdem Realität in einer Datenbank modellhaft abgebildet wird. Ohne Bezug dazu adressiert ›Relationales Modell‹ die Art und Weise der Datenorganisation im Speicher und dient der Unterscheidung zum ›CODASYL‹-Netzwerkmodell und zum ›hierarchischen Modell‹ der Datenspeicherung. Im Prinzip ließe sich in allen drei genannten Organisationsmodellen das gleiche Informationsmodell realisieren – auf jeweils eigene technische Weise. Zur besseren Unterscheidungen wird im Folgenden wenn vom relationalen Modell die Rede ist, die Wendung ›relationale Logik‹ eingesetzt.

Codd einen impliziten Hinweis, welchem informationstheoretischen Begriff dieser Ansatz verbunden ist. Jene Quellenangabe war in der folgenden, breit zirkulierenden Publikation von 1970 in den *Communications of the ACM* nicht mehr vorhanden, wodurch – aus heutiger Perspektive – die spätere Rezeption der relationalen Algebra als Singularität begünstigt wurde. Codd schrieb 1969: »Man stelle sich beispielsweise eine Datenbank, welche Informationen über Teile, Projekte und Anbieter enthält vor. Die einzelne Beschreibung für jedes einzelne Objekt (beispielsweise für ein spezifisches Teil) wird als Entität bezeichnet [Mealy 1967]« (Codd 1969, 2).<sup>xxxvii</sup> Um seine Verwendung von ›Entität‹ zu belegen, stellte Codd einen Verweis auf den Text *Another Look on Data* von George H. Mealy 1967 bei.<sup>52</sup> Diese Bezugnahme erscheint wie ein fehlendes Mosaikteil, denn sie diskutierte die Informations-Modellierung von Semantiken der Symbolverarbeitung.

Wie Mealys Titel *Another Look at Data* errahnen lässt, finden wir darin einen Hinweis, in welchem Sinne Konzepte wie ›Daten‹ oder ›Entität‹ von Codd verwendet wurden, aber auch darauf, dass Mealys Sichtweise auf den Begriff Daten sich von anderen, seinerzeit gebräuchlichen Konzepten unterschied. Als die Publikation erschien, war Mealy für IBM in Poughkeepsie führend an der Entwicklung des Betriebssystems OS/360 beteiligt und stieß in diesem Rahmen auf das Problem der Maschinenunabhängigkeit – Ziel der Entwicklung von OS/360 war ein Betriebssystem, welches auf verschiedenen Computern einer Modellreihe lauffähig war und modular angepasst werden konnte.<sup>53</sup> Die bis dahin von IBM entwickelten Betriebssysteme für verschiedenen IBM-Computer waren nur bedingt untereinander anschlussfähig, sodass Software nur eingeschränkt austauschbar war. Dies beinhaltete auch das Problem, dass die Form der Datenspeicherung verschieden sein konnte. Selbst, wenn ein Programm mit einigen Anpassung übertragbar war, mussten die dazugehörigen Daten häufig für verschiedene Figurationen aufwändig aufbereitet werden.

Mealy verzeichnete drei Ebenen der Datenverarbeitung:

- Wirklichkeit – reale Welt,
- Informationsmodell – Ideen, die in der menschlichen Wahrnehmung als Modell der Wirklichkeit existieren,
- Speicher – Symbole, die auf einem Speichermedium, ebenfalls in Form eines Informationsmodells der Wirklichkeit, abgelegt sind.

»Daten sind, so könnten wir sagen, Fragmente einer Theorie der wirklichen Welt, und Datenverarbeitung verändert die Repräsentationen dieser Theorie-Fragmente« (Mealy 1967, 525).<sup>xxxviii</sup> In Absehung

---

<sup>52</sup> Veröffentlicht als Beitrag zur *Fall Joint Computer Conference 1967* in Anaheim, Kalifornien und in der Folge in den *Proceedings of the ACM*.

George H. Mealy (1927–2010): Studium der Mathematik an der Harvard University bis 1951. Ab 1955 Professur für Mathematik und Computer Science an der Harvard University. In den Bell Labs entwickelte Mealy ab circa 1957 eines der ersten Betriebssysteme namens BESYS für den IBM 704, welches über die SHARE Nutzergruppen auch anderen Anwendern zur Verfügung gestellt wurde. Als Neuheit ermöglichte es das Eingreifen des Operators im laufenden Betrieb, das kontinuierliche Einlesen nacheinander folgender Lochkartenstapel, neue Formen des Zugangs für verschiedene Nutzer und Fehlerkontrolle und Debugging im laufenden Betrieb (Drummond, 1987). Im Jahr 1959 wechselte Mealy zur RAND Corp. wo er Datenstandards für Betriebssysteme erforschte und als Teamleiter das RAND-SHARE Betriebssystem für den IBM 7090 mitentwickelte (Bryan 1962). Nach der Arbeit an diesen stapelorientierten Betriebssystemen war er bei IBM an der Entwicklung der OS/360 beteiligt, welches erstmals stapelorientierte- und interaktive Arbeitsweisen ineinander vereinte und mehrere parallel verlaufende Prozesse erlaubte.

<sup>53</sup> Im Zuge der Arbeiten an OS/360 dürfte sich George Mealy auch mit Charles T. Davies ausgetauscht haben (siehe Abschnitt 2.1.4 Datenunabhängigkeit).

einer spezifischen Ontologie schlug Mealy vor, sich an die Nominalisten zu halten, eine Strömung der logischen Philosophie. Namentlich bezog er sich auf deren Vertreter Willard van Orman Quine.<sup>54</sup> Des- sen Ausführung ist Teil einer umfangreichen Diskussion, die sich seit Platon grob entlang folgender Unterscheidung entfaltet: In der Platonschen Auffassung existieren abstrakte Objekte als universelle Prinzipien. Es gibt demnach ›Rotes‹, oder ›die Menschheit‹ als allgemeine, abstrakte Prinzipien, die universelle Gültigkeit besitzen. Diese ›realistische‹ Strömung der Philosophie geht davon aus, dass die Gattung oder das Wesen der Existenz der Dinge und der Individuen vorausgeht.

Von den Vertretern des Nominalismus wird diese Sicht hingegen als ontologisierend kritisiert. Sie ver- weisen darauf, dass es sich um von Individuen geschaffene Begriffsbildungen handele. Demnach stellen Individuen die primäre Realität dar, aus der sich durch Abstraktion entsprechende Universalien ablei- ten lassen. ›Rotes‹ ist der Name – deswegen auch Nominalisten – für eine Gruppe von beobachtbaren Einzelobjekten. Realität komme nur Einzelobjekten zu. ›Rotes‹ sei keine unmittelbare Realität, sondern der Name einer Verallgemeinerung für eine Gruppe einzelner, konkreter, realer Gegenstände – eine Abstraktion die der menschliche Geist leistet, die nicht bereits ›von vornherein da ist‹. Quine stand – zumindest in dem Text auf den Codd Bezug nahm – in einer Reihe von Theoretikern wie Carnap und Wittgenstein, einer Sicht, demnach logische Klassen, Zahlen und Funktionen nicht eigenständig existie- ren (Quine 1953, 14). In diesem Sinne ist Bedeutung nicht zu verwechseln, mit dem Namen, den ein Objekt oder eine Klasse von Objekten erhält. Bedeutung ergibt sich nicht durch Schließen oder Reflek- tion, sondern durch empirische Beobachtung des Gegenstandes. »Freges Beispiel des ›Abendsterns‹ und ›Morgensterns‹, und Russels ›Scott‹ und ›der Autor von Waverley‹ zeigen, dass Begriffe dasselbe Ding benennen, aber in ihrer Bedeutung variieren können.<sup>55</sup> Die Unterscheidung zwischen Bedeutung und Benennung ist ebenso bedeutend auf der Ebene abstrakter Begriffe. Die Begriffe ›9‹ und ›die An- zahl der Planeten‹ bezeichnen ein und dieselbe abstrakte Entität, aber müssen mutmaßlich als ungleich in der Bedeutung angesehen werden; denn astronomische Beobachtung war Voraussetzung und nicht allein die Überlegungen über die Bedeutung, um die Gleichheit der fraglichen Entität festzustellen« (Quine 1953, 21).<sup>xxxix</sup>

Mealy schlug daher vor, sich einer Theorie von Relationen anhand eines Beispiels aus der Genealogie anzunähern, da die Frage der Familienabstammung wesentlich auf Beziehungen (engl.: Relations) be- ruhe. Genealogische Zusammenhänge ließen sich beschreiben als »Menge von Objekten, *Attribute* die- ser Objekte und *Werte der Attribute*«, zwischen denen Beziehungen bestünden (Mealy 1967, 526). Da- bei sei es so, dass diese nicht zwangsläufig hierarchisch abzubilden sind, denn die Kinder zweier Eltern konnten durch Trennung und Wiederheirat Kinder einer anderen Familie werden und nicht an eine einzige Familie angeknüpft werden. Vielmehr gelte: »... im Grunde sind Relationen Entsprechungen zwischen n-Tupeln aus einer Menge mit m-Tupeln aus einer möglicherweise verschiedenen Menge« (Mealy 1967, 526).<sup>xl</sup> Diese nicht-hierarchische Darstellung von Beziehungen ist auch eines der wesent- liche Merkmale der von Codd beschriebenen relationalen Algebra.

Mealy entwarf ein Konzept von Daten und Wirklichkeit, das diese mithilfe der klassischen Mengenthe- orie aufeinander abbildet, wobei er vor allem das Verhältnis von Entitäten, Werten und Struktur, also der Frage, wie sich Entitäten zu Werten verhalten und welchen Typs Werte sind, widmete. »Im ersten

---

<sup>54</sup> Im Vorwort zu *A Logical Point of View* (1953) dankt Quine unter anderen Rudolf Carnap, Alonzo Church, Nelson Good- man, Alfred Tarski und Morton White. Diese Mathematiker, Logiker und Philosophen seiner Zeit stecken den Referenzrah- men für seine eigenen Theorien dar. Quine selbst verabschiedet sich später vom strengen Nominalismus, den er in *A Logical Point of View* vertrat. Ein weiterer Bezug: Codd verwendet als Referenz für seine Relationale Algebra ein Lehrbuch von A- lonzo Church, welcher mit Quine in engem Austausch stand.

<sup>55</sup> Anmerkung zur Übersetzung: Der Roman *Waverley* von Walter Scott erschien 1814.

Teil dieses Aufsatzes haben wir ein theoretisches Modell von Daten und Datenverarbeitung vorgeschlagen. Das Modell ist ein System von Mengen aus Entitäten, Werten, Datenmaps und Prozedurenmaps. Die Entitäten korrespondieren mit den Objekten der wirklichen Welt, über welche Daten erhoben und gespeichert werden. Die Datenmap schreibt den Attributen der Entitäten Werte zu; diese Maps werden verstanden als Mengen aus geordneten Paaren von Entitäten und Werten beziehungsweise als Datenobjekte« (Mealy 1967, 533).<sup>xli</sup> Mit dieser Argumentation schlug Mealy ein abstrahiertes Konzept von Daten (als Datenobjekte) vor, welches sich vom späteren Vorschlag zur relationalen Algebra deutlich unterschied. Doch sind Einflüsse deutlich erkennbar:

- Erstens zeigte es auf, dass Daten mithilfe von Mengentheorie begriffen und bearbeitet werden können,
- zweitens wird deutlich, auf welche Weise Daten unabhängig von der unterliegenden maschinellen Struktur zu behandeln seien.

Dabei trennte Mealy zwischen Datenmaps, Strukturmaps und Operationen, »eine saubere Unterscheidung der strukturellen Fragen von anderen« (Mealy 1967, 529).<sup>xlii</sup> Datenmaps enthielten beispielsweise Personen oder Geburtsdaten, während Strukturmaps die Datenmaps untereinander verknüpfen, also beispielsweise die Verknüpfung von Person und Geburtsdatum beschrieben.<sup>56</sup> In der relationalen Algebra schaffen Relationen diese Verknüpfung.

Wie können in dieser Informations-Modellierung von Wirklichkeit in Relationen semantische Aspekte abgebildet werden? Quine argumentierte, dass die Semantik sich aufspalte in eine Theorie der Bedeutung und eine Theorie der Referenz. Diese, notierte er ganz nominalistisch, seien so grundsätzlich verschieden, dass sie nicht einmal gemeinsam Erwähnung finden sollten. »Semantik« wäre ein guter Name für eine Theorie der Bedeutung, wenn nicht einige der besten Arbeiten in der sogenannten Semantik, darunter [Alfred] Tarski's, zu einer Theorie der Referenz zählen würde« (Quine 1953, 130).<sup>xliii</sup>

Dem Feld der Bedeutung schrieb Quine zu:

- Synonymität (Ähnlichkeit von Bedeutung)
- Signifikanz (Vorhandensein von Bedeutung)
- Analytizität (Wahrheit aufgrund von Bedeutung)
- Implikation (Wenn-Dann-Relation)

Dem Feld der Referenz ordnet er zu:

- Namen/Benennung
- Wahrheit (im Sinne von Wahr/Falsch-Prädikatenlogik)
- Denotation (Referenzpotential)
- Extension (Ersetzbarkeit von Namen durch andere Namen)
- Werte in Variablen (Quine 1953, 130)

Diese Aufzählung kann uns bereits als Anhaltspunkt dienen, wie Mealy und im Anschluss Codd ihre Konzepte im Verhältnis zum Feld der Semantik anlegten. Sie bewegten sich im Feld der Referenz, nicht der Bedeutung. Quine konstatierte zwar, dass die nominalistische Sicht keineswegs alle Probleme der Logik, jene Momente der Logik, die mathematisch paradox sind, aus dem Weg räumen könne. Er

---

<sup>56</sup> Das was Mealy als Strukturmaps bezeichnet, ist zu vergleichen mit Pointern in der bis dahin üblichen Terminologie.

schließt jedoch: »Tarski's Konstruktion von Wahrheit ist leicht auf andere Konzepte einer Theorie der Referenz auszuweiten. Es ist ein beeindruckender Fakt, dass diese Begriffe, trotz der Paradoxa, die wir mit ihnen verbinden, deutlich weniger nebelig und unklar sind, als jene Begriffe, die zu einer Theorie der Bedeutung gehören« (Quine 1953, 138).<sup>xliiv</sup>

Wenn sich in Bezug auf die relationale Algebra die Frage der Semantik stellt, so bezieht sich diese auf Fragen der Referenz. Herauszuheben ist, dass im nominalistischen Modus Referenz nicht erkannt sondern zugeschrieben wird und diese Zuschreibung im Rahmen eines Informations-Modells erfolgt. Voraussetzung für diese Referenz bleibt, dass ein Objekt tatsächlich vorhanden und beobachtbar ist, Abstrakta können, wenn überhaupt, nur unter Bezugnahme auf die ihnen zugrundeliegenden Objekte formuliert werden.

Eine der relationalen Algebra vergleichbare Funktionalität entwarf Mealy nicht. Er stellte allerdings Operationen (bei Mealy: Prozeduren) als etwas dar, das in der Lage ist, die Datenmaps zu verändern. Bei beiden Autoren, bei Mealy und bei Codd, folgt daraus, dass sich aus in Mengen organisierten Daten durch mengentheoretische Logikoperationen neue Mengen generieren lassen, dass also die Neukombination zusätzliche Informationspotenziale erschließt.

Mealy argumentierte, dass die von ihm vorgeschlagenen Datenmaps eine abstrahierte Theorie der Wirklichkeit darstellen. Daher könne nicht Computer-Hardware für deren Verarbeitung maßgeblich sein, diese sei nicht abstrakt. Vielmehr sei es eine »abstrakte Theorie«, die sowohl in der Lage ist Datenverarbeitung zu modellieren, als auch Modelle der Wirklichkeit abzubilden, von Nöten. Datenverarbeitung solle also nicht allein unabhängig sein gegenüber dem Umstand, auf welchem Computer und mit welcher Sprache sie stattfindet – und auf diese Weise eine größere Austauschbarkeit zwischen den verschiedenen Betriebssystemen und Sprachen erlauben. Sie solle auch unabhängig sein gegenüber der Repräsentation von Daten, also gegenüber der logischen Struktur, in der diese Daten im Speicher organisiert sind. Bisher habe man Symbole als Namen von Speicherorten behandelt und nicht als Namen von Entitäten, die durch die Datenstruktur vergeben wurden (Mealy 1967, 525–529). Um dies zu erreichen, wird eine vermittelnde Schicht, er nannte sie »Representation Map«, eingezogen. Der Datenzugriff sei kein Problem der Organisation der Daten im Speicher, sondern ein Feature des Verarbeitens von Daten an sich. »Verschiedene Prozeduren werden, generell gesprochen, auf die gleichen Daten in unterschiedlicher Weise zugreifen wollen. Die Reihenfolge, in der die Datenobjekte abgerufen und gespeichert werden, ist (oder sollte) unabhängig von der Organisation der Daten...« (Mealy 1967, 530).<sup>xlv</sup> Codd griff die Frage der Ordnung als eine der Abhängigkeiten auf, die in seinerzeitigen Betriebssystemen existieren und denen die relationale Algebra abhelfen sollte (Codd 1970, 377). Die Entitäten in der Informationsmenge sind demnach dadurch bestimmt, dass sie beobachtbar einer Domäne angehören, die in der Datenbank durch einen Namen angesprochen werden können. In der relationalen Algebra, die eine Entität als Tupel fasst, ist es notwendig, dass alle Tupel einer Relation gleich aufgebaut sind, dass also von allen Entitäten die gleichen Attribute erfasst werden und dass Attribute immer vom selben Typ (Ganzzahl, Gleitkommazahl, Text usw.) sind. Dabei dürfen die Werte einzelner Attribute durchaus leer sein, sie müssen allerdings aufgeführt werden.

#### 2.1.2.2 Zusammenfassung

Shannons Schaltalgebra erlaubt es, Vakuumröhren, Transistoren, und Mikroprozessoren mit einer digitalen Logik des An und Aus, der 1 und 0 algebraisch zu verknüpfen. Sie zeigt auf, wie mittels

Schaltungen gerechnet werden kann und wie bestimmte logische Operatoren wie AND, OR, NOT als Schaltung zu realisieren sind. Auch Codd's relationale Algebra erlaubt diese Boolesche Logik zum Abgleichen der Prädikate. Codd konnte, so er denn nicht selbst auf diese Idee verfiel, bei Mealy sehen, wie auch der semantische Gehalt von Information in die Areale der Berechenbarkeit zu überführen ist, und zwar als Referenz. Dies bedeutet noch lange nicht eine computable Semantik, sondern den mit Hilfe der Logik und der Mengentheorie begründeten Vorschlag, den semantischen Gehalt *in Namen*, mit den computablen Daten verknüpft, parallel zu führen. In Codd's Lösung bedeutet das, ähnlich wie bei Mealy, dass Objekte als Entitäten beschrieben werden. Eine Entität umfasst nicht die vollständige Beschreibung eines realen Objektes, sondern ausgewählte Aspekte, welche *in Attributen* als *Werte* abgelegt werden. Die anderen Eigenschaften des Objektes, die in den Entitäten nicht erfasst werden, sind gleichsam abgespalten und der Computation entzogen. Sie werden nicht Teil des Modells von Wirklichkeit, welches Berechenbarkeit ermöglicht.

Codd besteht darauf, dass über die verschiedenen Objekte hinweg in den Domänen (später: Spalten), die Datentypen gleich bleiben, um deren algebraische Definiertheit zu gewährleisten. In den Namen, die für diese Domänen vergeben werden, ist der semantische Bedeutungsrest aufgehoben, der von der Maschine nicht, sondern vom Menschen nur verstanden und als Referenz erkannt wird. Mealy, der seine theoretische Intervention an nominalistische Positionen Quines gebunden hat, inspiriert Codd, Entitäten nominal zu beschreiben: Sie bezeichnen nicht ein abstraktes Konzept, sondern ergeben sich aus der konkreten Beobachtung der Objekte. Eine Domäne mit dem Namen ›Farbe‹ enthält also nicht etwas, was abstrakt für das Konzept Farbe an sich steht, sondern der semantische Gehalt dieses Namens bestimmt sich über die aufgeführten Werte der Entitäten, zum Beispiel ›rot, rot, gelb, rot, blau, violett‹ und so weiter. Erst die aufgeführten Werte erlauben eine Bestimmung der Referenz des Namens ›Farbe‹. Diese Nuance entgeht der Maschine Computer. Für Computer handelt es sich um Zeichenketten. Zum Einsatz kommt vielmehr dessen Fähigkeit in, für menschliche Verhältnisse, extremer Geschwindigkeit zu berechnen, ob die Entitäten in der Domäne ›Farbe‹ der Relation ›Farbschema‹ und der Domäne ›Farbe‹ der Relation ›Gegenstände‹ per relationaler Algebra, Übereinstimmungen oder Unterscheidungen im Sinne der Wahrheitswerte True und False ergeben.

Somit wird die relationale Algebra zu einem Interface zwischen maschineller Berechenbarkeit einerseits und der menschlichem Wahrnehmung der Wirklichkeit andererseits, welche allein in der Lage ist, eine Verständnis für den Zusammenhang von ›rot, rot, gelb, rot, blau, violett‹ und dem Namen ›Farbe‹ zu entwickeln.

## 2.1.3 Mengentheorie und Maschinen-Unabhängigkeit

### 2.1.3.1 Kontext: Timesharing und Virtuelle Adressierung

In der Verarbeitung großer Informationsmengen entstand mit Zunahme der Prozessorleistung ein Konflikt: Die vergleichsweise langsame Ein- und Ausgabe von Daten per Lochkarte oder Magnetband entwickelte sich zum Flaschenhals. Zwar schufen Festplatten, die nicht-sequentiell lesen und schreiben konnten, anfangs Abhilfe. Die Datenablage auf ihnen wurde jedoch gehandhabt, als ginge es um die altbekannten sequentiell zu lesenden oder zu schreibenden Speicher. Im Ergebnis hing die Lese- und Schreibgeschwindigkeit der Speicher der stetig steigenden Prozessorleistung permanent hinterher.



Moore's Law galt nicht für Massenspeicher. Wenn sich das Problem hardwareseitig nicht lösen ließe, gäbe es dann eine andere Lösungsmöglichkeit?

Im Gegensatz zu vielen der anderen Protagonisten dieser Geschichte war David Childs nur lose institutionell gebunden. Er arbeitete größtenteils von zu Hause aus. Der junge Mathematiker, der seinen Master in Logik, Philosophie und Computer Science absolviert hatte, war vom Projektleiter Franklin Westervelt engagiert worden, um am Projekt CONCOMP: Research in conversational use of computers an der University of Michigan teilzunehmen. Das von der Advanced Research Projects Agency (ARPA) finanzierte Projekt zum dialogorientiertem Einsatz von Computern verfolgte breit gestreute Anliegen, vor allem Grundlagenforschung zur Verarbeitung graphischer Eingaben.<sup>57</sup> Da die verfügbaren Betriebssysteme und Hardware den Anforderungen nicht gewachsen war, entwickelte die Universität auf Basis des vorhandenen IBM System 360 ein eigenes Timesharing System, das Michigan Terminal System, welches den gleichzeitigen Zugriff vieler Nutzer über Terminals erlaubte (Abb. 5).<sup>58</sup> Kennzeichnend für das Projekt war die zunehmend enge Verknüpfung von Hardware- und Softwareentwicklung, die ein Lerneffekt aus den Problemen der stark hardwareorientierten Computergroßprojekte der 1950er Jahre darstellte.<sup>59</sup>



Abb. 5: Login zum Michigan Terminal System auf einem IBM 3277 Display-Terminal. Ein Terminal hat keine eigene Rechenkapazität. Es dient lediglich der Darstellung der, über die Telefonleitung übertragenen, Daten, die vom Mainframe berechnet werden und der Dateneingabe, die direkt an den Mainframe übertragen werden, zum Beispiel den IBM 360/67 (CC-BY 3.0 Michigan Terminal System Archive).

<sup>57</sup> Hinzu kamen Echtzeit-Dialogsysteme, Sprachsynthese, Forschung an Betriebssystemen und informatorischer Logik, die Weiterentwicklung prozeduraler Computersprachen mit dem Schwerpunkt der Grafikverarbeitung.

<sup>58</sup> Im Abschlussbericht wurde die Zahl von 9000 Nutzern pro Tag angegeben (Westervelt 1970, 18). Erkenntnisse aus CONCOMP flossen später in die Entwicklung des regionalen Computer-Netzwerks MERIT und die Anbindung an das nationale ARPAnet ein. Die weiter entwickelte Programmiersprache MAD (Michigan Algorithmic Decoder) ermöglichte die Verwendung graphischer Datenelemente wie Punkt und Linien, und MAD führte Techniken zur modularen Erweiterbarkeit der Programmiersprache ein.

<sup>59</sup> IBM reagierte auf diese Problem, indem im Werk Poughkeepsie eine System Architecture Group an der gemeinsamen Konzeption von Software (vorrangig dem Betriebssystem) und Hardware für das Modell IBM 370 arbeitete. Eines der Mitglieder war Charles T. Davies, siehe nächster Abschnitt.

In diesem Umfeld war es David Childs Aufgabe, die mathematischen Grundlagen von Datenstrukturen zu erforschen. Die Anregung, sich in den Datenstrukturen nicht zuvorderst auf maschinen-orientierte Ansätze zu konzentrieren, stammte vom Projektleiter Frank Westervelt, der Child für das Projekt angestellt hatte. Aus Westervelts Sicht sollten die Nutzer zukünftiger Datenbankfigurationen, mit den erwartbar stark wachsenden Datenmengen, nicht mehr mit den Einzelheiten der Datenrepräsentation behelligt werden. Die Datenrepräsentation sollte vielmehr für die Nutzer transparent sein. »Dr. Westervelt hatte oft über die Notwendigkeit einer generellen Datenstruktur gesprochen, die nicht maschinen-, sondern informationsorientiert war, wobei der Designschwerpunkt auf den Informationsgehalt statt auf die Maschinenrepräsentation gelegt wurde« (Childs 1970, D6).<sup>xlvi</sup> Die Vermutung liegt nahe, dass Westervelt im Rahmen seiner Forschungstätigkeit an der University of Michigan eine Erfahrung gemacht hatte, die ihn von der Notwendigkeit der Maschinenunabhängigkeit überzeugte. Als möglicher Kandidat dafür kann ein Forschungsprojekt gelten, an dem Westervelt zuvor teilgenommen hatte.



Abb. 6: Michigan Terminal System, Mike Alexander an der Operator-Konsole. Hintergrund, rechts: Frontseite der IBM CPU des IBM System 360/67 (CC-BY 3.0 Michigan Terminal System Archive).

Er arbeitete mit einer Reihe anderer Wissenschaftler am Michigan Terminal System, welches den gleichzeitigen Zugriff auf die Ressourcen des universitären Großcomputers mittels Terminal ermöglichte (Abb. 6). In diesem Rahmen war er an der Entwicklung einer Methode beteiligt, die als Virtuelle Adressierung bekannt geworden ist.<sup>60</sup> War bis dahin der Nutzer eines Großrechners dafür zuständig, die Adressen der Speichersegmente, in denen ein Programm ablief oder Daten gespeichert wurden, selbst zu verwalten und im Prinzip diese Verwaltung der Speicheradressen direkt in ihrem Programm

---

<sup>60</sup> Virtuelle Speicheradressierung wurde an verschiedenen Institutionen parallel und größtenteils unabhängig voneinander entwickelt, so zum Beispiel das Atlas System/University of Manchester, der kommerziell verfügbare Burroughs B5000 oder der Rice Institute Computer/Houston, Texas. Die Speichervirtualisierung war eine notwendige Konsequenz von Mehrbenutzersystemen mit den damaligen Systemarchitekturen.

zu implementieren, so konnte dies im Time-Sharing beim gleichzeitigen Zugriff mehrerer Nutzer nicht länger funktionieren. Die Verwaltung und Adressierung des Speichers sollte nunmehr vom Betriebssystem vorgenommen werden. Dies gelang, indem die Programme einzelner User in kleine Einheiten aufgeteilt wurden, die automatisch auf verfügbare, adressierbare Speicherbereiche verteilt wurden (Westervelt u. a. 1966). Aus Sicht der Nutzer stand das Programm virtuell in seiner Gänze zur Verfügung, sie bemerkten die Segmentierung nicht. Diese virtuelle Speicherverwaltung zog eine zusätzliche Software-Schicht zwischen Nutzer und Maschine ein und ermöglichte den gleichzeitigen Ablauf unterschiedlichster Programme. Welchen Schluss mag Westervelt daraus gezogen haben? Offenbar jenen, dass es sich lohnt, auf der Softwareebene des Betriebssystems, den Nutzer von der eigentlichen maschinellen Konstitution zu distanzieren (Westervelt u. a. 1966, 3).

### 2.1.3.2 Maschinenunabhängigkeit und Relationale Mengentheorie

Childs – ebenfalls Nutzer des MTS – nahm die, von Westervelt angeregte, konzeptuelle Unterscheidung vor: Er betrachtete Informationen generell als maschinenunabhängig. Die Informationen, so Childs, würden erst zum Zwecke der Verarbeitung in die Maschinenrepräsentation »gezwungen«. Statt also danach zu suchen, wie die Datenstrukturen der Maschinenumgebung zu optimieren wären, suchte er nach einer Datenstruktur die der Informationsumgebung entspräche und sich an deren Logik orientierte.

Childs und Westervelt verstanden, dass sich bisherige Lösungsansätze in einem ersten Schritt auf einzelne Hardwarearchitekturen bezogen und erst danach im praktischen Einsatz als zweiten Schritt verallgemeinert wurden. Childs hingegen wollte die generellen Mittel für Informationsstrukturen zuerst erforschen und erst auf dieser Basis nach Mitteln der Umsetzung in der Maschinenrepräsentation suchen. »Das Problem liegt folglich darin, eine passende Struktur für die Informationsumgebung zu finden oder zu entwickeln, sodass die daraus ableitbaren Operationen das gleiche Ergebnis sicherstellen, unabhängig von jeglicher Maschinen-Repräsentation dieser Information« (Childs 1970, D-7).<sup>xlvii</sup> Hintergrund dieser Überlegungen war auch, dass Daten, selbst wenn sie im gleichen Speichermedium abgelegt sind, verschiedenen Logiken folgen: Während Software-Anwendungen auf Ausführbarkeit hin angelegt sind, folgen die zu verarbeitenden Daten einer anderen strukturellen Zugriffslogik. Die Antwort auf diese spezifische Zugriffslogik für Daten sah Childs in der Mengentheorie. Seine Hinwendung zur Mengentheorie erklärt Childs retrospektiv so: »Das Konzept eines ›Datensatzes‹ (als eine Sequenz von Bits oder Bytes) für die Repräsentation von ›Daten‹ war damals auf breiter Basis üblich. Die naheliegendste mathematische Abbildung eines Datensatzes war ein ›n-Tupel‹ in der Mengentheorie. Damit schien die Mengentheorie als guter Ausgangspunkt, um mathematische Grundlagen von ›Datenstrukturen‹ zu untersuchen« (Childs 2016b, per E-Mail).<sup>xlviii</sup> Dieser Ansatz unterschied sich wesentlich von den sonstigen Konzepten zur Datenorganisation, denn er orientierte sich an einer eigenen Logik, unabhängig von maschinellen Strukturen. Die Forschung war seinerzeit durch drei Publikationen beeinflusst: Thoralf Skolems Aufsatz *Two remarks on set theory* (Skolem 1957) und John McCarthys Artikel *A Basis for Mathematical Theory of Computation* (McCarthy 1961), welche mathematisch-logische Probleme bei der Verwendung von n-Tupeln aufzeigten und Patrick Suppes' Buch *Axiomatic Set Theory* (Suppes 1960), anhand dessen Childs das binäre Relationenkonzept zu n-nären Relationen mathematisch korrekt erweiterte (Childs 2016c per E-Mail). Mit der Idee einer mengentheoretischen Datenstruktur verabschiedete sich dieses Konzept von fixierten Strukturen, bei denen Zeiger auf eine bestimmte Speicheradresse direkt verwiesen und diese Adressen fest in Programme eingeschrieben

waren. Mengentheoretische Operationen, wie zum Beispiel UNION und INTERSECTION,<sup>61</sup> ersetzen jene fest programmierten Verweise die bis dahin üblicherweise durch Zeiger auf physische Adressen erfolgten. Boolesche Operatoren erlaubten die Abbildung einer Prädikatenlogik auf die Mengen, zum Beispiel: »A ist eine Untermenge von B; A ist gleich B; A und B sind disjunkt« (Childs 1968, 5).<sup>xlix</sup> Jede Anfrage war nunmehr als ein mengentheoretisches Problem zu formulieren. Dabei ist jede einzelne Menge von anderen Mengen unabhängig. Dies erlaubte, jede Menge zu verändern, ohne dass eine andere Menge oder die Gesamtstruktur davon beeinflusst wird, ein zentrales Moment des Strebens nach Datenunabhängigkeit.

Die Verschachtelung und Kombination verschiedener Mengen ist potentiell unendlich, nur begrenzt durch den zur Verfügung stehenden Speicher. Die Frage anhand derer erst festgestellt wird, welche Mengen überhaupt abgefragt werden können, und welche Operationen innerhalb und zwischen den Mengen durchgeführt werden sollen, kann nunmehr ad-hoc gestellt werden und muss nicht mehr in die Programmstruktur eingeschrieben sein. Childs: »Eine mengentheoretische Datenstruktur ist im Grunde eine Metastruktur, die es einer Frage erlaubt, die Struktur des Datenflusses zu ›diktieren‹. Eine Frage etabliert, auf welche Mengen zugegriffen wird und welche Operationen innerhalb und zwischen den Mengen ausgeführt werden« (Childs 1968, pt. Abstract).<sup>1</sup> Die Fragestellung ist nicht der Struktur untergeordnet, wie bis dahin üblich, sondern die Fragestellung selbst erzeugt erst eine Struktur, die jedoch nur für den Moment der Frage existiert.

Childs entwickelte somit eine Theorie von Datenrelationen, auf die er eine mengentheoretische Operationen anwandte – mit Sicherheit ein wesentlicher Hinweis für die relationale Algebra. Codd kannte Childs Forschung, weil er durch seinen Ph.D. 1965–1967 an der University of Michigan mit den universitären Forschungsumfeld in Verbindung stand. Im Zuge der Arbeit an der relationalen Algebra standen beide Autoren lose in Kontakt, da Childs per Brief einige Fragen Codd's zu seinen Veröffentlichungen beantwortete (Childs 2016c, per E-Mail).

Aus Childs Sicht bestand das Problem darin, dass Entwickler die Daten in Datensätzen verankert sahen und in dieser Logik verharrten. Ihm ging es darum, sich davon zu lösen und Daten als mathematische Objekte zu sehen. »Es war durchaus akzeptabel für die Suche nach Daten mathematische Konzepte einzusetzen, aber die Modellierung von Daten als mathematische Objekte war ›Ketzerei‹« (Childs 2016c, per E-Mail)<sup>li</sup> Nach Ende der CONCOMP Förderung 1970 gründeten Westervelt und Childs die Firma Set Theoretic Information Systems (STIS). Es gelang, ein Speichermanagementsystem, welches die Daten in einer mengentheoretischen, mathematischen Speicherorganisation abbildete, in den produktiven Einsatz zu bringen, und zwar auf den damals üblichen IBM Großrechnern. Die Office of Research and Development des U.S. Department of Labor entwickelte beispielsweise ab 1970, basierend auf Childs Speicherzugriffstechnologie, eine Datenbankanwendung namens MICRO, die unter anderem zur Auswertung statistischer Daten des Arbeitsmarktes diente und bis 1998 eingesetzt wurde.<sup>62</sup> Diese

---

<sup>61</sup> Die 1968 von Childs vorgeschlagenen Operationen lauten vollständig: »Union, Intersection, Symmetric Difference, Relative Complement, Exactly N elements of A, DOMAIN of A, RANGE of A, IMAGE of B under A, CONVERSE IMAGE of B under A, CONVERSE of A, Restriction of A to B, RELATIVE PRODUCT of A and B, CARTESIAN PRODUCT of A and B, DOMAIN CONCURRENCE of A relative to B, RANGE CONCURRENCE of A relative to B, SET CONCURRENCE of A relative to B, CARDINALITY of A« (Childs 1968, 5).

<sup>62</sup> Vgl. *MICRO Information Management System – Reference Manual* (Kahn/Rumelhart/Bronson 1970) und *On the Feasibility of a Labor Market Information System – Final Report for Period July 1, 1970-June 30* (Cohen 1974); In den Folgejahren entwickelte Childs seine Extended Set Theory (XST) weiter und versuchte, sie auf breiter Basis in Anwendung zu bringen. Childs beobachtet, dass in der klassischen Mengentheorie und deren Umsetzung in SQL nur ein Teil der tatsächlich möglichen mathematischen Operationen der Mengentheorie implementiert wurde. Er schlägt vor, den Ballast der Operationen in

Entwicklung begann im gleichen Jahr wie die Entwicklung des IBM Peterlee Relational Test Vehicle und drei Jahre vor den ersten Arbeiten an System R, und INGRES, führte jedoch nicht zu einer nachhaltigen Beeinflussung anderer Softwareprodukte.

Childs sah die Grundlagen für den Erfolg der relationalen Algebra in der Reduzierung – jedoch nicht der Aufhebung – der Abhängigkeit zwischen Nutzeransicht der Daten und der Maschinenrepräsentation der Daten. Der Soziologe Michael Castelle hat darauf hingewiesen, dass beide – Childs und Codd – aus ihrer mathematischen Perspektive dazu neigen, die Prozessualität der Datenhaltung und -verarbeitung, also die Notwendigkeit, dass bestimmte Schritte nacheinander erfolgen, vergleichsweise niedrig zu bewerten. Das ist ein Verweis darauf, warum sie einerseits aus der Position der Mathematiker heraus in der Lage sind, einen innovativen Ansatz für den Zugriff auf Information zu entwickeln, aber andererseits die Überführung der Innovation in die informatorische Realität von Buchhaltung, Wissensmanagement und Produktionsprozessen durch andere popularisiert wurde.

### 2.1.3.3 Zusammenfassung

Childs und Westervelt schlugen kurz vor Codd ein neues Paradigma des Informationszugriffes vor: statt Informationen in Datensätzen zu organisieren und den Speicherort der Datensätze auf dem Speichermedium direkt anzusteuern, sollen Informationen in n-Tupeln organisiert werden, die mit den mathematischen Funktionen der Mengenlehre identifiziert werden können. Die Nutzer setzten sich allein mit der mathematischen Repräsentation auseinander, die Übersetzung in physische Adressen übernahm das relationale Datenbankbetriebssystem.

Der mathematisch begründete Zugriff auf Informationen verändert außerdem die Zugriffsweise zur Informationsanalyse: Mussten zuvor Informationen aus verschiedenen Dateien ausgelesen werden und für ihre Verarbeitung spezifisch zusammengestellt, vorformatiert und abgespeichert werden, so wird nun der statistische Blick auf Informationen erleichtert. Beispielsweise können Summen oder Sortierungen unmittelbar auf die Informationsmengen ausgeführt werden, da diese mengentheoretisch ohne Zwischenschritte verarbeitbar sind.

Wenn durchaus kritisch die Unsichtbarkeit und Intransparenz algorithmischer Prozesse zu konstatieren sind, so können wir retrospektiv beschreiben, wie der unmittelbare Zugriff auf Informationen in diesem Zuge dem Blick der Nutzer entzogen wurde. Techniken wie die virtuelle Adressierung oder der relationale Datenbankzugriff verbergen absichtlich Fragen der Adressierung vor den Nutzern. So verdeutlichte Codd 1970 »da mehr Nutzer gleichzeitigen Zugang zu einer großen Daten Bank erhalten, geht die Verantwortung für effiziente Antwortzeiten und Informationsfluss vom einzelnen Nutzer zum Datensystem über« (Codd 1970, 381f.).<sup>63</sup> Der Einzug einer zusätzlichen algorithmischen Schicht abstrahiert, erleichtert und verschleiert die ›maschinellen Verhältnisse‹.<sup>63</sup>

Warum aber musste Codd überhaupt ins Spiel kommen, um den von Childs entwickelten mengentheoretischen Ansatz als ›relational‹ zu popularisieren? Auch Codd hatte deutliche Probleme, sich bei IBM mit seiner Idee durchzusetzen, einerseits gegen vorhandene IBM Datenbankprodukte, wie

---

relationalen Datenbanken, wie zum Beispiel die Erstellung von Indexen und die Verwendung von Caches zu streichen, und mittels XST zu lösen, siehe auch <http://xsp.xegegis.org>. Den Unterschied zwischen einer mathematisch fundierten Vision und der tatsächlichen Umsetzung in relationalen Datenbank-Anwendungen kennzeichnet Childs so: »Relationales Datenmodell: mathematisch zuverlässig, deklarativ, Verarbeitung von Mengen; Relationales Datenbank Management System: physisch abhängig, prozedural, Verarbeitung von Datensätzen« (Childs 2016d).

<sup>63</sup> Die absichtliche Verschleierung von Adressen ist sonst nur bekannt aus dem historischen Kriegsgeschehen, bei dem Straßenschilder abmontiert und Hausnummern entfernt worden, um dem Feind einen Teil der Orientierung zu nehmen.

beispielsweise IMS, andererseits gegen Ingenieure, die einen maschinenorientierten Ansatz verfolgten.<sup>64</sup> Doch insgesamt war seine institutionelle Verankerung bei IBM, einem der größten Computer- und Softwareproduzenten jener Zeit hilfreich in der öffentlichen Wahrnehmung seiner Ideen.<sup>65</sup> Ihm mag zu Gute gekommen sein, dass sein Proposal in größeren Magazinen publiziert wurde als Childs Theoreme, in den *Communications of the ACM* und direkt von Dritten aufgegriffen wurde, so von den Gründern von Oracle oder Ingres. Childs war für die University of Michigan tätig, allerdings nur lose institutionell verankert. Somit war er auf Frank Westervelt angewiesen, sein Konzept innerhalb der Institution zu verankern. Dies gelang offensichtlich nur zum Teil, beide gründeten eine Firma aus, um ihre Idee weiter zu entwickeln.

Die Mathematiker Edgar F. Codd und David Childs eint, dass beide in mathematischen Konzepten dachten. Codd stellte seine relationale Algebra ursprünglich in Form von Arrays dar, bevor er dem Hinweis von Don Chamberlin, dem Co-Entwickler von SQL, folgend, das visuelle Interface der Tabelle einsetzte. Childs Extended Set Theory XST erfährt ihre Darstellung in XML, einer Datenstruktur, die durchaus auch Nicht-Mathematikern zugänglich sein kann. Jedoch fehlt ihr die visuell-diagrammatische Qualität räumlich organisierter Information. Dieser Umstand stellte sich einer weiteren Popularisierung ebenso in den Weg, wie die angesprochenen institutionellen Unterschiede.

#### 2.1.4 Datenunabhängigkeit

Edgar F. Codd und Charles T. Davies lernten sich vermutlich im Rahmen ihrer Tätigkeit für IBM in Poughkeepsie, New York State kennen. Im Gebäude 701 nördlich des Hauptgebäudes, welches sich in Ziegelsteinbauweise über 800 Meter Länge erstreckte, befand sich die Abteilung zur Software Entwicklung mit ca. 500 Mitarbeitern. Codd war bereits länger in Poughkeepsie tätig. Er leitete nach seiner Rückkehr aus Canada, ein Team, das jenen Teil des Betriebssystems für den IBM Stretch Computer (später IBM 7030) programmierte, der die gleichzeitige parallele Ausführung verschiedener Programme erlaubte – damals eine neue Technologie. Im Anschluss war Codd mit einem IBM-Stipendium an die Universität Michigan gewechselt, um dort seinen Ph.D. abzulegen. Ab 1966 war er wiederum in Poughkeepsie im Bereich Software für IBM tätig.

Davies, der zuvor Logistikproblem der US Airforce in Dayton, Ohio löste, arbeitete ab 1966 in Poughkeepsie an der Systemarchitektur des IBM System/370. In dieser Abteilung war Davies für Programmierung, also die Umsetzung dessen, was das Planungskomitee entwickelt hatte, zuständig. Systemarchitektur bedeutete, dass Hardware und das Betriebssystem möglichst eng ineinander verzahnt entwickelt werden sollten. Der bis dahin übliche Modus, zuerst die Hardware-Entwicklung abzuschließen und dann die Software zu entwickeln, zog teils absurde Umwege in der logischen Struktur des Betriebssystems nach sich.

---

<sup>64</sup> »Die CONCOMP-Veröffentlichungen waren nicht geheim, aber auch nicht frei verfügbar. Obwohl Codd die Artikel Childs zitierte, ist unklar, wie viele Leute damals in der Lage waren, über Childs Forschung zu lesen. Die CONCOMP-Papiere trugen den Vermerk, dass sie allein für »geeignete Personen« verfügbar sein sollten« (North 2010). Auch heute wird Codd von seinen ehemaligen IBM Kollegen als singular dargestellt und Childs Leistung übergangen, zum Beispiel heißt es, »Ted Codd's grundlegender Vorschlag für eine neue Herangehensweise an Datenbankmanagement ...« in (Wade/Chamberlin 2012, 40).

<sup>65</sup> Das Selbstverständnis der IBM war in den 1960ern geprägt von der Hardwareproduktion, und sah sich in der langen Tradition der Büromaschinenproduktion. Letztlich produzierte IBM jedoch auch signifikante Softwarepakete, zum Beispiel die Betriebssysteme, die an den Verkauf der Hardware gekoppelt waren. 1968 wurde IBM von den Kartellbehörden gezwungen, die Software von der Hardware abzukoppeln und einzeln zu verkaufen, um so den Markt für Wettbewerber zu öffnen. Dies führte zur Erkenntnis, des Wertes von Software (Ceruzzi 2003, 106f.).

Sharon Codd schildert die Bekanntschaft in einem Interview: »Er wurde durch einen Freund bei IBM auf das Datenbankproblem aufmerksam. Dieser Mann war nicht nur ein Freund, sondern er war in New York auch ein Nachbar. So kam es, dass sich Ted [Codd] ansah, was daran beteiligt war, und entschied, die Mengentheorie darauf anzuwenden. Sein Name war Charles Davies. Es ist allerdings bemerkenswert, dass nachdem Ted an Relationen und relationalen Datenbanken gearbeitet hatte, Davies sagte, dass Ted seine Ideen gestohlen hätte. Aber alles, was Ted machte, war die Anwendung von Mathematik. Die ganze Sache beinhaltete Mathematik, daher war Davies' Behauptung ziemlich albern« (Interview Sharon Codd 2016).<sup>liv</sup>

Ein übliches Problem dieser Zeit bestand darin, dass die auf Magnetband sequentiell abgelegten Daten weniger schnell eingelesen werden konnten, als der Rechner sie eigentlich verarbeiten konnte. Ein ›Operator‹ war für das Einlegen des passenden Bandes, für das Zurückspulen und die Lagerhaltung der verschiedenen, damals vergleichsweise teureren Bänder zuständig. War der Operator nicht im richtigen Moment zur Stelle, konnten Pausen eintreten, und die Rechenmaschine stand still. Dies war eines der Probleme an denen Davies Arbeitsgruppe forschte. Die Problemlage war ihm bereits länger bekannt, denn in den frühen 1960ern setzte er sich im Rahmen einer Arbeitsgruppe für das Air Force Logistics Command am IBM 7080 mit der Fragestellung der rechtzeitigen Ein- und Ausgabe auseinander. Die Anwendung, die auf diesem Rechner lief, betraf ein Inventar von 200.000 Einzelteilen, sodass sich die Arbeitsgruppe bereits früh mit Fragen großer Daten-Banken konfrontiert sah. In Poughkeepsie arbeitete Davies vor diesem Erfahrungshintergrund zusammen mit Codd an Fragen der Datenspeicherung. Kent Salmond, ein Kollege von Charles Davies berichtet: »Ted [Codd] war zu diesem Zeitpunkt in Poughkeepsie [1966/67] und trieb sein Flat-File-Konzept voran, dass alles aussehen sollte wie ein Flat-File. Und Charles arbeitete mit ihm daran. Damals gab es dort keine große Auswahl an Datenbanksystemen, aber es gab Dateisysteme, und sie arbeiteten an Dateisystemen« (Interview Salmond 2016).<sup>lv</sup> Die Datenstruktur der Flat-Files bezog sich in ihren Ursprüngen auf die Lochkarte in diesem Falle also IBMs Standard mit 80 Spalten und 10 Zeilen. Üblicherweise wurden mehrere Datensätze des gleichen Typs zu einer Datei zusammengefasst. Die Datensätze besaßen immer die gleiche Größe, so wie jede Lochkarte dieselbe Speichermenge erlaubte.<sup>66</sup> Jeder Datensatz bestand aus einer Reihe von Feldern, in die Werte eingeschrieben werden konnten und welche immer die gleiche Reihenfolge besaßen. Aus dieser regelmäßigen Struktur folgte die Bezeichnung *Flat File*. Deutlich sehen wir hier wie sich die Organisation der Daten an der vorhandenen Materialität orientiert und nicht an der logischen Struktur von Informationen. Ein Vorteil der Flat-Files bestand allerdings darin, dass die Datenrepräsentation einfach verständlich, in tabellarischer Form, möglich war, und dass einzelne Entitäten (Zeilen) über ID-Schlüssel angesprochen werden konnten.

#### 2.1.4.1 Datenunabhängigkeit

Codd benannte 1969 das Ziel, Programmstrukturen und Datenstrukturen voneinander zu trennen, so dass eine Trennung der sprachlichen Logik des Zugriffs auf Daten von der maschinellen Adressierungslogik im Speicher möglich wird. Zu diesem Zeitpunkt handelte es sich um konzeptuelle Überlegungen für eine zu schaffende Figuration – um Grundlagenforschung. In vorhandenen Figurationen, wie IBMs IMS oder SDCs TDMS sieht Codd mit der Verwendung von Datenbeschreibungs-Tabellen

---

<sup>66</sup> Vgl. ausführlich *Data Base Technology* (McGee 1981, 506).

(»data description tables«) einen Weg zur Datenunabhängigkeit eingeschlagen, jedoch nur in Ansätzen, da drei Arten von Datenabhängigkeit nach wie vor bestünden

- Als Beispiel für Sortierungsabhängigkeiten nennt er erstens eine Stückliste in einer Datei. Diese Stückliste war für gewöhnlich nach Artikelnummer absteigend sortiert und abgespeichert. »Anwendungsprogramme, die sich auf die abgespeicherte Sortierung einer Datei beziehen, laufen Gefahr nicht mehr korrekt zu arbeiten, wenn es aus irgend einem Grund nötig wird, diese Sortierung durch eine andere zu ersetzen« (Codd 1970:378).<sup>lvi</sup>
- Zweitens vermerkte Codd für die Abhängigkeit der Indexierung vermerkte Codd, dass diese, um die Performance des Datenzugriffs zu erhöhen, Daten redundant repräsentiere. Mit sich verändernden Daten, müssten auch die Indizes neu aufgebaut werden, woraus sich die Frage ergab, ob Softwareanwendungen und der Datenzugriff über ein Time-Sharing-System es erlaubten, von diesen Änderungen ungestört weiter zu funktionieren.<sup>67</sup>
- Drittens ging es um die Abhängigkeiten der Zugriffspfade: In Netzwerk- oder hierarchischen Modellen waren die Anwendungsprogramme von deren logischer Struktur abhängig. Bei Änderungen traten zwangsläufig Probleme auf, denn in einer Netzwerkstruktur gab es zwischen den einzelnen Knoten immer ein Eltern-Kind-Verhältnis der Informationsknoten und daraus resultierend Abhängigkeiten zum logisch übergeordneten Knoten.

Dem hoffte Codd mit seinem Vorschlag abzuhelfen, mit einer mathematisch begründeten »relationalen Sicht auf Daten«. Diese mathematisch-logisch begründbare Sicht auf Daten, führte in die bisher vom Maschinellen ausgehende Debatte einen neuen Abstraktionsgrad ein, welcher sich nicht länger an den physischen Aspekten der Datenspeicherung auf einem Datenträger orientierte.

Zwei Jahre davor, 1967, veröffentlichte Charles Davies IBM-intern eine Studie zum Datenzugriff über symbolische Pointer, im Unterschied zu den damals üblichen physikalischen Pointern. Symbolische Pointer basierten darauf, dass die einzelnen Entitäten in einer Datei durch ID-Schlüssel angesprochen werden konnten, um zueinander gehörige Entitäten zu verbinden. Damit konnte man sich von den durch die Hardware vorgegebenen physikalischen Pointer emanzipieren. Hier schimmert bereits das Potenzial der Datenunabhängigkeit durch.

In dem 1970er Papier bedankte sich Codd im letzten Abschnitt, kurz vor den Quellennachweisen: »Es war C[harles] T. Davies von IBM Poughkeepsie, der den Autoren von der Notwendigkeit der Datenunabhängigkeit in zukünftigen Informationssystemen überzeugte« (Codd 1970, 387).<sup>lvii</sup> Codd erwähnte jemanden, der ihn »überzeugte«, jedoch ohne Textquelle blieb. Das Überzeugen fand nicht schriftlich, sondern in den Arbeitsräumen, den Fluren oder in der Kantine in Poughkeepsie statt. Codd war nicht von Anfang an der Meinung, das Daten, maschineller Layer und Programmlayer voneinander getrennt werden müssten. Jemanden überzeugen, bedeutet einen längeren Prozess des Gesprächs, den Austausch von Argumenten, einen produktiven Dissens, der sich in eine Richtung hin auflöst.

Wir wissen nicht genau, mit welcher Argumentation Davies Codd überzeugte, aber es deutet sich in Davies' Arbeiten an (Davies 1967; Davies 1973; Davies 1978). Und wir können als Zeugen Cods Text aufrufen, den wir danach befragen, wie er das Problem der Datenunabhängigkeit anging. Gleich im zweiten Absatz wurde das Problemfeld umrissen: »[...] die hier behandelten Probleme sind jene der

---

<sup>67</sup> Die in der relationalen Logik angestrebte Indexfreiheit war in den frühen Implementierungen relationaler DBMS nicht verwirklicht und ist es auch heute in der Mehrzahl nicht – IBMs System R und die Sprache SQL, das an der Universität Berkeley entwickelte Ingres und auch die ersten relationalen Datenbanken der Firma Oracle nutzten Indexierung für schnellere Suchergebnisse. Sie wurden dafür von Codd kritisiert, der darin ein ungültige Vereinfachung des von ihm verfolgten relationalen Calculus sah.



Datenunabhängigkeit – der Unabhängigkeit der Softwareanwendung und Terminalaktivitäten von der Zunahme der Datentypen und Änderungen der Datenrepräsentation – und bestimmten Arten von Dateninkonsistenzen« (Codd 1970, 377).<sup>lviii</sup> Codd selbst musste die Problemlage einige Jahre zuvor in ähnlicher Form wie Davies begegnet sein, während seiner Arbeit für das IBM Projekt Stretch, denn der gleichzeitige Ablauf mehrerer Programme bedurfte der ständigen Adressverwaltung im Speicher – sowohl der Programme selbst, als auch der Daten. Im Abschlussbericht des Projektes Stretch formulierte er bereits die Notwendigkeit einer gewissen Datenunabhängigkeit: »Die Anordnung von Platz im Kernspeicher und Plattenspeicher, die Zuweisung der Eingabe/Ausgabe-Einheiten, und die Kontrolle des Time-Sharing sollte auf den Bedürfnissen der ausgeführten Programme basieren (und nicht auf zufälligen, rigiden Unterteilungen der Maschine)« (Codd u. a. 1962, 194).<sup>lix</sup> Codd war für die Fragestellung sensibilisiert. Davies musste ihn nur überzeugen und lieferte wertvolle Vorarbeiten mit der Idee der symbolischen Pointer in Flat-Files.<sup>68</sup> Eine überraschende Wendung nahm die Beziehung zwischen Codd und Davies, als weitere Beteiligte ins Spiel kamen – David Childs und Frank Westervelt von der University of Michigan/Ann Arbor. Childs erinnert sich: »Es scheint, dass Frank Westervelt das Konzept der ›Datenunabhängigkeit‹, welches die Essenz des ›relationalen Konzepts‹ darstellt, unmittelbar initiiert hat. Kein anderer kümmerte sich damals um diese Frage. Sogar Codd erwähnt, dass er die Idee 1967 von Charles Davies, der mit Frank Westervelt zusammen arbeitete, übernommen hat und ich [David Childs] wurde auch direkt von Frank 1965 darauf aufmerksam gemacht« (Childs 2016a, per E-Mail).<sup>lx</sup>

Wann und unter welchen Umständen trafen Westervelt und Davies aufeinander, wie es Childs berichtet? Wir erinnern uns, dass Davies ab 1966 für IBM Poughkeepsie in der Programmierung der Systemarchitektur IBM System 370 arbeitete, dem Nachfolger jenes Betriebssystems, das durch Westervelt und seine Kollegen an der University of Michigan für das CONCOMP-Projekt eingesetzt und zu einem Multi-User Timesharing-System, IBM System 360/67, erweitert wurde (Westervelt u. a. 1966). Im Abschlussbericht des CONCOMP Projektes stellen Westervelt et. al. klar, dass das von IBM versprochene Time-Sharing-System nicht rechtzeitig und nicht im entsprechenden Funktionsumfang geliefert wurde, und sie daher ihr eigenes Michigan Terminal System entwickelten. Die Erkenntnisse daraus feedbackten sie an IBM zurück, die das IBM 360 System, darauf basierend, zum IBM 360-67 weiter entwickelten. Das Bild, demnach Frank Westervelt die Unabhängigkeit von Datenstrukturen gegenüber den zugrunde liegenden maschinellen Strukturen, wesentlich ideell begründete, verdichtet sich. Davies war ab 1971 für den IBM-Standort Santa Theresa in der General Product Division tätig und entwickelte in den folgenden Jahren das Konzept der Spheres of Control Einfluss, welches auf die Fähigkeit von Datenbanken, Transaktionen abzubilden, hatte.

#### 2.1.4.2 Zusammenfassung

Frank Westervelt spielte an der University of Michigan/Ann Arbor, an der Edgar Codd und David Childs studierten bzw. arbeiteten, und mit der Charles Davies interagierte, eine wesentliche Rolle darin, jene Ideen zu propagieren, welche die Speicherung von Daten unabhängig von der zugrundeliegenden maschinellen Struktur organisieren sollten.

---

<sup>68</sup> Der Übersichtsartikel *Data Base Technology* (McGee 1981) weist darauf hin, dass innerhalb von IBM an mehreren voneinander unabhängigen Stellen am Flat-File Konzept mit Mengen aus Entitäten gearbeitet wurde und nennt: C. T. Davies *A Logical Concept for Control and Management of Data*, 1967; N. Raver *File Organization in Management Information Control Systems*, 1968; H. S. Meltzer *Data Base Concepts and Architecture for Data Base Systems* 1969; R. W. Engles *A Tutorial on Data Base Organization* 1972.

So verschiebt sich der bisherige Focus der Darstellung auf IBM, zugunsten eines Wechselverhältnisses zwischen der Bildungsinstitution Michigan State University, die im CONCOMP Projekt durch die DARPA gefördert wurde, und dem Büro- und Rechenmaschinenhersteller IBM und dessen Grundlagen- und Anwendungsforschung.

## 2.1.5 Die relationale Sicht auf Informationen und normalisierte Beziehungen

»In der relationalen Algebra waren die grundlegenden Objekte Tabellen, und man kombinierte diese Tabellen mit Operationen wie Join oder Projektion und ähnlichem. Der relationale Calculus war irgendwie eine seltsame Notation mit vielen Quantoren darin« (Don Chamberlin in: McJones 1997, 11).<sup>lxi</sup>

»Das relationale Modell besteht aus einer Sammlung von Relationen, die jeweils als einfache Tabelle gedacht werden können« (Michaels/Mittman/Carlson 1976, 128).<sup>lxii</sup>

### 2.1.5.1 Menge

Die relationale Algebra ist in einem mathematischen Feld begründet, welches zum Zeitpunkt ihrer Veröffentlichung kaum 90 Jahre alt war – der Mengentheorie. Als angewandter Mathematiker gelang es Codd, dieses Feld der Mengentheorie mit Hilfe des Konzeptes der Relation in die Welt der Rechenmaschine zu übertragen. Dazu genügt ihm neben seinem im Studium und der Berufstätigkeit erworbenen Wissen ein Buch, welches der Mathematiker Alonzo Church für die Lehre geschrieben hatte und das 1956 durch die Princeton University Press unter dem Titel *An Introduction to Mathematical Logic* veröffentlicht wurde. In der Einführung auf verweist Church in einer Fußnote auf die zugrundeliegenden Arbeiten von Gottlob Frege und Bertrand Russel und legt eine Spur in die Geschichte der formalen Logik, der wir, entlang des Begriffes der Menge, kurz folgen wollen. Das Konzept der Menge nach Cantor ist wesentlicher Teil der mathematischen Formalisierungsbestrebungen im Zeitraum zwischen 1870 und 1910. Mit diesen Formalisierungsbestrebungen sind neben dem, in Halle angesiedelten Mathematiker Georg Cantor, auch die Arbeiten des in Jena tätigen Mathematikers und Logikers Gottlieb Frege zur Prädikatenlogik befasst. Ziel war es, die »»intuitiven« Erklärungen und Argumente, die zu den Elementen der Mathematik gehören, durch formale Beweise, basierend auf logisch präzisen Definitionen oder Systemen von Axiomen« zu ersetzen (Bunn in: Grattan-Guinness/Bos (Hrsg.) 2000, 220).<sup>lxiii</sup> Diese Bemühungen kulminierten in dem, von Bertrand Russel und Alfred N. Whitehead 1910 herausgegebenen, mehrbändigen *Principia Mathematica*.

»Unter einer ›Menge‹ verstehen wir jede Zusammenfassung  $M$  von bestimmten wohlunterschiedenen Objecten  $m$  unsrer Anschauung oder unseres Denkens (welche die ›Elemente‹ von  $M$  genannt werden) zu einem Ganzen« (Cantor 1895, 481). Mit dieser Definition eröffnete Cantor das Feld der Mengentheorie. In seinem Aufsatz demonstriert er die Potentiale von Mengen, indem er unter anderem deren Vereinigung, Teilmengen, Äquivalenz aufzeigte. Bereits zwölf Jahre zuvor, 1883, hatte er unter dem Begriff der Mannichfaltigkeitslehre in einer Fußnote die Menge im mathematischen Sinne definiert: »Unter einer Mannichfaltigkeit oder Menge verstehe ich nämlich allgemein jedes Viele, welches sich als Eines denken lässt, d. h. jeden Inbegriff bestimmter Elemente, welcher durch ein Gesetz zu einem Ganzen verbunden werden kann...« (Cantor 1883, 43). Cantor band dieses Ganze zurück an Platos Dialog

zwischen Sokrates, Protarchos und Philebos und das dort begründete Konzept des *μικτόν*, des Gemischen, und schrieb der Menge einen holistischen Charakter, als das Ganze aller Ideen zu.<sup>69</sup> Wichtiger aus einer Perspektive des relationalen Datenmodells ist jedoch seine Bezugnahme auf *εἶδος* und *idéa* – Begriffe, welche die Gestalt oder Form eines Gegenstandes ansprechen, jedoch nicht als unmittelbar Gesehenes, sondern als im Geiste Modelliertes und dadurch Gesehenes. Diese im Geiste konstruierte Sichtbarkeit wiederum bezog sich auf gemeinsame Merkmale, welche durch Klassifizierung erkennbar und einer spezifischen Gesamtheit zuordenbar sind. Auf diese Weise positioniert er seine ›Mannichfaltigkeitslehre‹ als Antwort auf die bereits Plato plagende Frage danach, wie das Eine in der Lage sei, die Elemente des Vielen zu vereinen.

Der Philosoph Paul Livingston situiert Cantors Mengenlehre an der Membran des Universellen und Partikularen, denn die Beziehung einer Menge zu ihren Elementen verweise auf die Beziehung des Universellen und der darin eingeschlossenen Individuen. »Somit kann die Definition einer Menge als formal identisch verstanden werden zur Definition eines Konzepts oder eines allgemeinen Begriffs. Diesem Ansatz entsprechend, erblicken wir in der Struktur von Mengen und der ansonsten undefinierten Relation  $\in$  die tatsächliche zugrunde liegende formale Struktur einer Beziehung, die zwischen einem Objekt und einem Prädikat gilt oder einer Eigenschaft, für die geltend gemacht werden kann, dass sie dafür gilt. Das ist nichts anderes als die Beziehung zwischen dem Universellen und dem Partikularen [...]« (Livingston 2011, 44).<sup>lxiv</sup> Die in der mathematischen Logik formalisierten Mengen waren Ausgangspunkt für die Überlegungen zur relationalen Algebra, doch entzündete sich ihre Verwendung an einem anderen, der informatorischen Praxis entstammenden dringendem Problem.

### 2.1.5.2 Modell, Schema, Relation

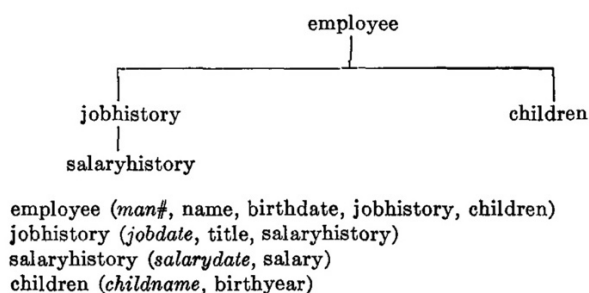


FIG. 3(a). Unnormalized set

Abb. 7: Aufbau von Domänen und Relationen im hierarchischen Modell, wie es in IBMs IMS, oder Bachmanns IDS eingesetzt wurde. ›Job history‹ und ›children‹ sind nicht-vereinfachte Domänen der Relation ›employee‹. ›Salary history‹ ist eine nicht-vereinfachte Domäne von ›jobhistory‹ (Codd 1970, 381).<sup>70</sup>

Das obige Beispiel (Abb. 7) setzte Codd ein, um die Funktion und die Nachteile des Netzwerkmodells zu demonstrieren. Schrieb man ein Programm, das durch diese Datenstruktur navigierte, so begann

<sup>69</sup> Das Gemischte entsteht, indem ἄπειρον, d.h. das Unendliche oder das Unbegrenzte, und πέρας, die Begrenzung, miteinander in Beziehung gesetzt werden. Das Unbegrenzte ist dabei nie völlig zähmbar und dieser prekäre Umstand kann zum Misslingen der Mischung führen (Mouroutsou 2010 S. 281). Zur πέρας, der Begrenzung oder dem Eingegrenzten, zählen hingegen Gegenstände die als bestimmbare Größen voneinander abgrenzbar sind und damit mathematisch ausdrückbar sind. πέρας ist in der Lage ἄπειρον einzuhegen, wodurch auch letzteres in den Bereich der Zahlen gerät. Dies ist dann das Gemischte, μικτόν. Für Cantor ist diese Diskussion vor allem wichtig, da er transfinite Zahlen als Extension der reellen Zahlen positionieren will.

<sup>70</sup> Domäne war in der Veröffentlichung im Sinne von ›Datentyp‹ beziehungsweise ›Spalte‹ zu lesen (Date 2015, 7f.).

dessen Prozedur bei ›employee‹, um dann zu ›jobhistory‹ und ›salaryhistory‹ weiter zu schreiten. Ein direkter Zugriff auf ›salaryhistory‹ war nicht möglich. Diese ineinander verschachtelte Organisation von Daten im Netzwerkmodell – mit einer Hierarchie aus Elternelementen (hier: ›employee‹) und Kinderelementen (hier: ›jobhistory‹, ›children‹) – erforderte eine große Anzahl prozeduraler Anweisungen, welche die Informationen erschlossen und zusammenfassten. Daher suchte Codd eine Möglichkeit der Vereinfachung. Er evaluierte eine Reihe von Konzepten, darunter auch den von David Childs entwickelten mengentheoretischen Ansatz und Levien/Marons Organisation von Informationen in Tupeln. Die Informationen, schlussfolgerte er, könnten so reorganisiert werden, dass sie als Mengen repräsentiert würden, die durch relationalen Calculus der Berechenbarkeit unterlägen. Dabei ging es auch um eine ›Verflachung‹ (›Flatness‹) der Struktur, die es erlaubte, jene Prozeduren des Hinabsteigens in den Informations-Hierarchien zu minimieren, durch die sich das Netzwerkmodell auszeichnete. »Es scheint lohnenswert, die Möglichkeit zu untersuchen, wie nicht-vereinfachte Domänen<sup>71</sup> eliminiert werden können. Es gibt, in der Tat, eine sehr einfaches Vorgehen zur Eliminierung, welches wir als Normalisierung bezeichnen werden« (Codd 1970, 381).<sup>lxv</sup> Warum war Codd diese Erkenntnis so wichtig? Die Normalisierung war Voraussetzung für einen relationalen Calculus. Die Anomalien in Relationen sollen beseitigt werden, um eine ›normale Form‹ zu erreichen. Dieses Vorgehen entstammt mathematischen Denkweisen, in denen es darum geht, ein mathematisches Objekt in eine gewünschte Form zu überführen, die dessen weitere Operationalisierung erlaubt.

Der relationale Calculus und die relationale Algebra stellten die Voraussetzungen dar für ein vergleichsweise einfache Abfragesprache, die sich signifikant von hierarchischen Strukturen unterschied, deren navigatorischer Zugang allein von hoch-spezialisierten und in die Struktur eingearbeiteten Programmierern zu bewältigen war.<sup>72</sup> Codd hatte die Problematik während seiner Tätigkeit in Poughkeepsie an IBM Information Management System (IMS), einer hierarchisch organisierten Datenbank, hinreichend kennen gelernt.

Der relationale Calculus beinhaltete für Codd ein weiteres Versprechen. Der navigatorische Zugang des CODASYL-Modells erforderte zahlreiche Zeilen prozeduralen Programmiercodes. Mit dem relationalen Calculus konnten diese auf ein Minimum reduziert werden und weckten die Hoffnung auf eine nicht-prozedurale, eine deklarative Figuration, die Hoffnung auf eine weitgehende Eliminierung der Prozedur.<sup>73</sup> »Codd zeigte einige ziemlich komplizierte Abfragen, und da ich den CODASYL-68-Standard untersucht hatte, konnte ich mir vorstellen, wie diese Abfragen in CODASYL durch Programme, die fünf Seiten lang sind, und durch ein Labyrinth von Pointern und ähnlichem navigieren, dargestellt würden. Codd konnte sie sozusagen als Einzeiler aufschreiben« (Don Chamberlin In: McJones 1997, 10).<sup>lxvi</sup>

---

<sup>71</sup> Entlang von Dates Argumentation wäre das Zitat wie folgt zu paraphrasieren: »Es gilt, die Möglichkeit zu untersuchen, wie nicht-vereinfachte Typen eliminiert werden können. Das Vorgehen dazu nennt sich Normalisierung«, siehe auch (Codd 1990, 20; Darwen 2012, 33).

<sup>72</sup> Der Calculus ist vor allem wichtig, da er mathematisch begründet und validiert werden kann, was mit relationaler Algebra allein nicht möglich ist. So sind für relationale Systeme Validierungen der Algorithmen und auch Performance-Vergleiche möglich, die mit früheren, nicht-formalisierten Systemen nicht möglich waren.

<sup>73</sup> In der Implementierung von SQL sind Codd's Hoffnungen eines komplett deklarativen Ansatzes nicht verwirklicht. Edgar Codd und Chris Date zeigten sich davon enttäuscht und kritisierten prozedurale Umsetzungen des relationalen Modells zeitweise sehr nachdrücklich (Darwen 2012, 32). Chris Date and Hugh Darwen entwickelten diese Kritik ab Anfang der 2000er Jahre weiter zu einem *Third Manifesto* (Date/Darwen 2000).

<i>supply</i>	<i>(supplier</i>	<i>part</i>	<i>project</i>	<i>quantity)</i>
	1	2	5	17
	1	3	5	23
	2	3	7	9
	2	7	5	4
	4	1	1	12

FIG. 1. A relation of degree 4

Abb. 8: Diagrammatische Abbildung einer Relation vierten Grades in Form einer Tabelle in der Veröffentlichung von 1970. Allerdings sprach Codd damals noch nicht von der Tabelle sondern von einem Array (Codd 1970, 381).

In Veröffentlichungen von 1970 und 1971 war keine Rede von Tabellen (Abb. 8). Die Daten waren im relationalen Calculus gefasst und stellen die mathematisch als Mengen gedachten Informationen als Array dar. Dieser Array nahm in Druckform zwar eine tabellarische Form an, aber Codd enthielt sich der Verwendung des Begriffs ›Tabelle‹ aus mathematisch-formalen Gründen. Er sprach von einem Array, wobei er bereits einzelne Elemente in den Termini von Tabellen beschrieb: »Diese Interface-Darstellung beruht auf einer rechteckigen Array, mit den folgenden Eigenschaften: 1.) Innerhalb jeder Spalte ist der Daten[typ] homogen; 2.) verschiedene Spalten können unterschiedliche Datentypen haben; 3.) Alle Zeilen bestehen aus distinkten Tupeln« (Codd 1971, 41).<sup>lxvii</sup>

Dass die tabellarische Darstellung der Relation eine spezifische alltagspraktische Qualität darstellte, bemerkten mit einiger Verzögerung Dritte. Sie erkannten, dass gerade die Tabelle ein geeignetes User-Interface für die relationale Mengentheorie darstellte, eine Membran zwischen »Anschauung und Denken« (Krämer 2010), die durch ihre kulturtechnische Vertrautheit einen Zugang zum relationalen Modell eröffnen konnte. Von Codd nicht intendiert, ermöglichte die Tabelle das Verständnis von relationalen Datenbanksystemen, ohne die komplexen mathematischen Hintergründe zu kennen, also auch für Nicht-Mathematiker. Das Dilemma lag darin, dass eine Tabelle nicht exakt eine Relation ist. Beispielsweise etabliert eine Tabelle durch die übliche Leserichtung eine An-Ordnung der Spalten (von links nach rechts) und eine An-Ordnung der Zeilen (von oben nach unten). In der mathematischen Definition der Relation hingegen gibt es keine Links-nach-Rechts-Ordnung oder Oben-nach-Unten Rangfolge (Interview Date 2017, Fol 28).

Den Vorschlag, die Tabelle als Denkwerkzeug einzusetzen, machten Boyce und Chamberlin,<sup>74</sup> die in Reaktion auf Codds Vorschlag, drei Jahre später, die Abfragesprache SQUARE entwickelten. »Eine normalisierte Relation kann als Tabelle von n Spalten und einer variablen Anzahl von Zeilen gesehen werden« (Boyce u. a. 1973, 31).<sup>lxviii</sup> In einer E-Mail bestätigte Don Chamberlin, dass er und Ray Boyce erstmals die Tabelle in den Vordergrund ihrer Überlegungen zur Relationalen Algebra legten: »Ray und ich waren keine Mathematiker. Wir wollten eine Weg finden, um den Zugang zu relationalen Daten für Menschen ohne formale mathematische Ausbildung zu vereinfachen« (Chamberlin 2016, per E-Mail).<sup>lxix</sup>

Zum Hintergrund dieser diagrammatischen Verschiebung gehört, dass in den 1960ern der Bedarf an Programmierern derart stark gewachsen war, so dass viele Entwickler angestellt wurden, die keine

<sup>74</sup> Don Chamberlin (\*1944) schloss einen Ph.D. in Electrical Engineering ab und Ray Boyce einen Ph.D. in Computer Science. Chamberlin verfügt über ein Nebestudium im 1961 als Teil des Mathematik-Departments neu eröffneten Studiengang Computer Science an der Stanford University. Boyce/Chamberlin arbeiteten damals im IBM Forschungszentrum Yorktown Heights (NY) und wurden kurz nach der Veröffentlichung ihrer Ausarbeitungen zur SQUARE-Abfragesprache nach San Jose berufen, wo sich Codd bereits aufhielt. Chamberlin wurde Projektleiter jener Gruppe, die mit System R für IBM die erste tatsächliche Implementation einer relationalen Datenbank umsetzte. In diesen Rahmen entwickelte er eine Fortführung der auf dem relationalen Calculus basierten SQUARE, die einfachere relationale Algebra der Abfragesprache SEQUEL, die heute als SQL der Weltstandard relationaler Abfragesprachen ist.

formale Bildung in diesem Bereich hatten. »Das Problem ist, dass viele von ihnen weder Ingenieure, Mathematiker oder irgendwie wissenschaftliche bewandert waren. Ich traf English-Studenten, Leute die Geschichte studiert hatten, aber keine Mathematiker« (Interview Sharon Codd 2016)<sup>lxx</sup> beschreibt Sharon Codd die Situation retrospektiv. Sie war selbst als Mathematikerin ausgebildet und arbeitete für IBM, bevor sie sich gemeinsam mit Edgar Codd und Chris Date in den 1980ern mit einer Consulting-Firma selbständig machte. Edgar Codd, dem es Mitte der 1970er schwer fiel, für seinen Ansatz Unterstützer zu finden, wandte sich letztlich an die Universitäten, wo er durch Vorträge auf sein Konzept aufmerksam machen wollte. »Dort fand er Menschen, die verstanden, was er vorhatte. Und der Grund dafür war, dass diese Leute Mathematiker waren« (ebd.). Aus dieser Perspektive ist verständlich, warum Ted Cods Vorschlag anfangs mit Skepsis begegnet wurde. Die tabellarische Darstellung und der Übergang von einem relationalen Calculus zur relationalen Algebra werden als Faktoren der Durchsetzung des relationalen Modells häufig übersehen und sind einer neuen Bewertung zuzuführen. Wesentlichen Anteil daran hatten Don Chamberlin und Ray Boyce, deren Abfragesprache SQL eine bewusste Vereinfachung der mathematischen Notation Cods darstellte, unter Hinweis darauf, dass auf dem Computerkeyboard bestimmte Zeichen, die für den relationalen Calculus benötigt wurden, nicht vorhanden waren.

Chamberlin erinnert sich in einem Oral-History-Interview: »Die Ideen sprachen Praktiker nicht unmittelbar an, ich denke, größtenteils, weil Ted sie in mathematischen Symbolen und Terminologie formulierte. In seinen ursprünglichen Abfragesprachen verwendete er mathematische Notation wie zum Beispiel Universalquantor und Existenzquantor, und er benutzte eine Menge griechischer Buchstaben. Das erzeugt den Anschein, dass etwas sehr esoterisch und schwierig zu verwenden ist. Hingegen hatte er eigentlich vor, das Schreiben von Abfragen einfacher zu machen, nicht schwerer. Ich denke also, die Entwicklung einer Sprache, die auf englischen Stichworten basierte, welche man auf einem Keyboard tippen, und welche man intuitiv lesen und verstehen konnte, war ein Durchbruch, der es für viele einfacher machte, die innewohnende Einfachheit von Teds Idee zu begreifen. Das vereinfachte nicht die Ideen an sich, aber sie sahen schlichtweg einfach aus« (Chamberlin 2001, 9).<sup>lxxi</sup>

<b>Mathematisches Konzept</b>	<b>Informatorisches Konzept</b>
Relation R, n-ten Grades	Tabelle R mit Spaltenanzahl n
Kardinalität der Relation	Zeilenanzahl
Attribut	Spalte
Tupel	Zeile
Domain	Datentyp

Tabelle 1: Übersetzung der mathematischen Begriffe in informatorische Begriffe nach (Codd 1990, 20).

Mathematisch gesprochen, handelt es sich bei einer Relation um eine Menge an Tupeln, repräsentiert als Zeilen der Tabelle. Daraus folgt für die Gültigkeit einer Relation, dass die darin vorhandenen Tupel verschieden sind und keine Duplikate in der Relation enthalten sind. Tupel, die eine Relation konstituieren, haben den Vorteil, dass ihre Position in der Menge keine Rolle spielt, sie können unsortiert sein. Die Tupel/Zeilen können in der Reihenfolge ihres Einganges gespeichert werden, ohne dass dies Auswirkungen auf den Informationsgehalt hat. Damit kann das Hinzufügen oder Löschen einzelner Tupeln die Gesamtordnung nicht stören, denn die maschinell verordnete Logik der Reihenfolge von Ein- und Ausgang wird entkoppelt.

In der relationalen Algebra ist es valide, dass in mehreren Tupeln ein bestimmtes Attribut gleich sein kann, vorausgesetzt es gibt mindestens ein anderes Attribut, das die Tupel voneinander unterscheidet. Das folgende Beispiel ergibt eine Relation, da sich unter anderem durch die Schlüsselwerte die Tupel voneinander unterscheiden.

Relation: Songs

Schlüssel	Titel	Genre
324	A Hard Day's Night	Pop
325	A Hard Day's Night	Rock

Abb. 9: Die Schlüssel unterscheiden die Tupel. Auch das Attribut ›Genre‹ unterscheidet die Tupel. Für die pragmatische Anwendung mit Hilfe einer Abfragesprache ist diese Form jedoch nicht wünschenswert. Zum Beispiel ist es hier schwierig anzugeben, dass ein und derselbe Song in mehrere Genre gehört (Autor).

Obiges Beispiel lässt sich auch als verschachtelte Tabellen darstellen:

Relation: Songs

Schlüssel	Titel	Genre
324	A Hard Day's Night	Pop
		Rock

Abb. 10: Für Nutzer sicherlich verständlich, in der relationalen Algebra jedoch unerwünscht: Verschachtelte Relationen. Die Attribute (Guitar-Pop, Rock) in der Spalte Genre sind im Zuge der Normalisierung so aufzulösen, dass nur ein Wert vorhanden ist (Autor).

Eine ›Normalisierung‹ in der relationalen Algebra bedeutet, sich wiederholende Attributwerte zum Gegenstand jeweils eigener Relationen zu machen. Die Beziehung wird expliziert. Sie wird über den Schlüssel in einer Tabelle (z.B. Titel:Schlüssel) und gleichen Fremdschlüssel (im Beispiel unten: 255) in einer anderen Tabelle (z.B. Titel-Genre:Fremdschlüssel.Titel) hergestellt.

Relation: Titel

Schlüssel	Titel
254	Help!
255	A Hard Day's Night
256	...

Relation: Genres

Schlüssel	Genre
1	Pop
2	Techno
3	Rock

Relation: Titel-Genres

Fremdschlüssel.Titel	Fremdschlüssel.Genre
255	1
255	3
...	...

Abb. 11: Die Normalisierung und Aufgliederung von sich wiederholenden Attributen in jeweils eigene Relationen ermöglicht die Abfrage über eine Prädikatenlogik. So kann dargestellt werden, dass ein- und derselbe Titel zu mehreren Genres gehört (Autor).

Die Entschachtelung und ›Verflachung‹ der Datenstrukturen, von Codd als Normalisierung bezeichnet, sorgte dafür, dass Beziehungen als Schlüssel explizit gemacht werden konnten. Dies erlaubte überhaupt erst die Operationalisierung der Beziehungen mithilfe des relationalen Calculus. Codd legte die Beziehungsgeflechte offen und vereinfachte sie maximal. Statt einer Baumstruktur, die notwendigerweise aus einer Hierarchie von Eltern-und-Kinder-Knoten bestand, wurden Informationen in der Fläche ausgebreitet. Um im Beispiel zu bleiben: Die Nutzer könnten unmittelbar, deklarativ, die Relation ›Genre‹ abfragen, ohne übergeordnete Knoten ansteuern zu müssen.

Gleichzeitig wurde der Bedeutungszusammenhang jeder einzelnen Relation geschärft, da Referenzen, und zwar jene, die kategorisiert werden konnten – ausgedrückt dadurch, dass sie als Attribut bzw. Spalte erscheinen – voneinander getrennt wurden. Das relationale Konzept verlangt nach einem Informationsschema, das Trennungen (in verschiedene Relationen) explizit macht. Dabei erscheinen die Beziehungen zwischen diesen getrennten Relationen als neue Relation, in denen die Beziehungen als Tupel (Zeilen) organisiert sind (Michaels/Mittman/Carlson 1976, 128).

Da sowohl die Informationen als auch die Beziehungen zwischen den Informationen als Tupel niedergelegt sind, kann die Datenbankstruktur fast beliebig erweitert werden. Die Normalisierung von Daten ist als konstruktiver Prozess zu lesen, denn die Verwendung von Schlüssel-Fremdschlüssel-Tupeln erzeugt eine stärkere Offenheit neue Relationen hinzuzufügen, als dies im hierarchischen oder im Netzwerkmodell der Fall war.<sup>75</sup> Dies gilt für das Datenbankschema, die Struktur der Datenbank. Jenseits dieser constructio des Schemas ergeben sich neue Informationspotenziale durch die Abfragestruktur, die durch Relationen ermöglicht wird. So kann eine Relation sich durch Beziehungen (Schlüssel-Fremdschlüssel-Tupel) über viele Mengen (Tabellen) spannen und je nach laufender Abfrage neue Ergebnis-Mengen ergeben. Diese Ergebnis-Mengen entstehen nur für den Moment der Abfrage, sie werden von der Datenbank momentan kombiniert und ausgeliefert. Nach der Laufzeit der Abfrage existieren diese Mengen in der Datenbank nicht mehr. Die Datenbank wird so zu einer temporal begrenzten »Menge von Mengen« (Mackenzie 2010, 8)<sup>lxxii</sup>, auf deren Elemente sich mittels relationaler Algebra zugreifen lässt.

Spezifisch für Codd's Ansatz ist, dass in Relationen im Unterschied zur klassischen Mengentheorie die Tupel (Zeilen) ungeordnet sind, d.h. ihre Reihenfolge keinerlei Aussage trifft. Dies ist sicher einer der Gründe, warum Codd anfangs gern auf die Tabelle als Darstellung verzichtete, denn diese impliziert eine visuelle Ordnung der Zeilen von oben nach unten. Ebenfalls ungeordnet sind die Attribute (Spalten), welche von Codd als Domänen bezeichnet wurden und über Namen angesprochen werden.<sup>76</sup>

Diese Nicht-Ordnung ermöglichte eine programmiertechnische Flexibilisierung, denn Informationen konnten über ihre Elemente, z.B. ID-Schlüssel, angesprochen werden, nicht über ihre Anordnung. Das Problem der Anordnung war und ist, dass ein Programm sozusagen ›wissen‹ muss, ob die Anordnung im Sinne der eigenen Programmlogik korrekt ist. Beispielsweise muss ein Programm, das auf eine Datei (bestehend aus Namen und Alter) mit Informationselementen zugreift, ›wissen‹ ob die Datei sortiert nach Name oder nach Alter abgespeichert ist. Umfangreiche Sortieroutine sollten dies sicher stellen, aber auch Zählroutinen, denn die Anzahl der Elemente in der Datenmenge musste beständig mittels

---

<sup>75</sup> In den 2000er Jahren wurde diese am Schema orientierte Constructio als zu rigide und unflexibel im Umgang mit großen Datenmengen auf verteilten Rechnerfarmen erkannt. Mit Big Data, d.h. den Technologien Big Table, Hadoop und anderen wurden offenere Strukturen geschaffen, die nicht mehr dem ACID-Paradigma entsprechen.

<sup>76</sup> Codd's Auffassung dazu änderte sich mit der Zeit. Anfangs gab er an, dass die Domänen (Spalten) zu ordnen seien, jedoch durch Permutation untereinander austauschbar seien. Dies hing auch mit dem unklaren Status der Benennung der Domänen zusammen. Mit den ersten Implementationen schälte sich heraus, dass die Spalten auch über Namen angesprochen werden können und damit ist die Ordnung der Domänen unwesentlich.



Programmiercode überprüft werden. Dies ist in der Relation nicht mehr nötig, da die Position in der Menge irrelevant ist. Eine ständig auftretende Fehlerquelle wurde so eliminiert. Dies bestätigt retrospektiv auch Hugh Darwen: »Uns sprach in diesem Zusammenhang besonders an, dass weder die Attribute noch die Tupel einer Relation einer Ordnung unterlagen« (Darwen 2015, 2).<sup>lxxiii</sup> Dies scheint nur ein kleiner Unterschied, doch in Bezug auf Datenspeicherung war die Idee grundlegend neu. Die Schärfung und Aufspaltung verlangt von den Nutzern eine spezifische, informatorisch orientierte, Wahrnehmung der Realität. Wobei jene Aspekte, die der informatorischen Wirklichkeit nicht inhärent sind, abgespalten werden, und aus Sicht des Datenbankschemas nicht existieren. Die Realität wird entlang von Beziehungen und algebraischen Operationen in eine maschinenverarbeitbare Form gebracht, gerade weil im Datenbankschema vom semantischen Gehalt der Zeichen abgesehen wird, und der Computer die Informationen nur entlang der korrekten Syntax und jenseits eines pragmatischen Verständnisses verarbeitet.

Die relationale Algebra wendete sich explizit an die Nutzer, und nicht an die Programmierer großer, geteilter Datenbanken. Idealerweise ermöglicht sie Abfragen von Büroarbeitern, Bibliothekaren und Managern. Hier deutet sich eine Demokratisierung des Informationszugriffes an. Es scheint, als habe sich seine Intention heute eingelöst, wenn auch anders als von ihm intendiert. Es bedarf weiterhin spezialisierter Programmierer, um relationale Datenbanken abzufragen, jedoch hat sich der generelle Gebrauch von Datenbankmanagementsystemen seit den 1980er Jahren massiv ausgeweitet, und neue Formen und Ökonomien der Informationsverarbeitung nach sich gezogen.

### 2.1.5.3 Operationen

Die Ansprüche der Nutzer an ein Datenbanksystem leiteten sich auch aus jenen Figurationen ab, welche bereits verfügbar waren. IBMs Innensicht konzentrierte sich in den 1970er Jahren auf das Produkt IMS, welches aus den Report-Generatoren hervorgegangen war.<sup>77</sup> In Codd's relationaler Algebra stellten sich genau jene Berechnungen als fehlend heraus, welche den durch die Reportgeneratoren bereits erreichten Produktivitätsstand aufgreifen konnten. Die relationale Algebra sah anfangs keine Möglichkeiten vor, Zwischen- oder Teilsummen zu bilden, die in Berichten dringend benötigt wurden. Diese Aufgabe wurde an die Hostsprache, in welche die relationale ›Sublanguage‹ eingebettet werden sollte, delegiert. Auch das Gruppieren von Informationen mittels ›Group By‹ war ursprünglich nicht vorgesehen und wurden erst später in SQL durch Chamberlin et. al. verfügbar. Ähnlich verhält es sich mit Operationen, die den benötigten Diagrammatiken von Berichten zuarbeiten würden. Seitenumbruch oder die Erzeugung von Leerzeilen, um Informationen im Bildschirm- und Papiausdruck zu strukturieren, fehlten. Damit delegierte Codd den prozessualen Anteil aus der ›flächigen‹, deklarativen relationalen Algebra in die ›hügelige‹ Landschaft prozeduraler Programmiersprachen.

Hier entfaltet sich ein Spannungsfeld, welches zentral für die Entstehung der relationalen Algebra ist: Der Versuch einer theorie-geleiteten, mathematischen Fundierung des Konzeptes und das aus der Empirie der Alltagspraxen abgeleitete Interesse an einer bestimmten Form von Funktionalität, die auf eine möglichst breite Anwendbarkeit abzielt. Diese Spannung betrifft auch die Operationalität, die auf zweierlei Art beschrieben werden kann: als (mathematische) Operationen auf Mengen einerseits und im Sinne der Datenverarbeitung in DBMS andererseits. Die relationale Algebra basiert auf algorithmischen Verfahren der klassischen Mengentheorie, welche um einige wichtige Spezifika erweitert wurde. Aus der Binnensicht der Mathematik war dies nicht weiter spannend, ja fast banal – es stellte keine

---

<sup>77</sup> Eine Genealogie der Reportgeneratoren leistet *Evolution of Data-Base Management Systems* (Fry/Sibley 1976).

mathematische Neuerung dar. Codd jedoch verstand sich als angewandter Mathematiker, der zum Ziel hatte, mit Hilfe von Mathematik, Lösungen für informatorische Problem zu finden. Für die maschinenorientierten Ingenieurskulturen stellten seine Vorschläge eine wesentliche Neuerung dar.

Ein Teil der Operationen wurde unmittelbar aus der klassischen Mengentheorie übernommen. »Da Relationen Mengen sind, sind alle üblichen Operationen auf sie anwendbar« (Codd 1970, 383).<sup>lxxiv</sup> Der Reichtum verfügbarer Operationen wurde anfangs durch Codd nicht ausreichend herausgestellt. Den Lesern der 1970er Veröffentlichung konnten all jene Operationen entgehen, die in der klassischen Mengentheorie eine umfangreiche Kombinatorik ermöglichten, z.B. die Operanden  $>$ ,  $<$ ,  $=$ , oder die Verknüpfungen der Prädikatenlogik.<sup>78</sup>

Ursprünglich hatte Codd in der vorhergehenden, 1969 erschienenen, Publikation bei IBM diese Qualität stärker herausgestellt: »Solche Ausdrücke werden aus einfachen Namen der Relationen, relationalen Operatoren wie zum Beispiel  $=$ , logischen Konnektoren und den Quantoren des Prädikatenkalküls konstruiert« (Codd 1969, 10).<sup>lxxv</sup> Sehen wir uns die einzelnen Punkte dieser Aufzählung etwas näher an: Erstens, »logische Konnektoren«: Die klassische Mengentheorie hätte im Prinzip erlaubt, statt logischen Konnektoren mengentheoretische Operationen zu verwenden, wie Hugh Darwen, der bei IBM an einer relationalen Software, dem Business System 12 (1978–1985) arbeitete, retrospektiv anmerkt: »Wir stellten fest, dass alles was durch logische Konnektoren ausdrückbar war, einem äquivalenten Ausdruck mittel Ausschluss und/oder Schnittmenge, und/oder Vereinigung entsprach, aber wir glaubten nicht, dass dies für die Anwender akzeptabel sei« (Darwen 2015, 3).<sup>lxxvi</sup> Es zeigte sich in der tatsächlichen Implementierung von Abfragesprachen, auch wenn es Mathematiker als die elegantere Lösung ansehen, dass dieser Grad von Abstraktion für die angestrebten Endnutzer, die nicht unbedingt in mathematischer Logik gebildet sind, fern lag.

Nicht	Umkehrung
Oder	Alternation
Falls	Implikation
Falls ... Dann ...	Bedingung
Und	Konjunktion

Tabelle 2: Konnektoren der Prädikatenlogik nach Whitehead und Russels Principia Mathematica in dem von Codd herangezogenen Lehrbuch zur Mathematischen Logik von Alonzo Church (Church 1956, 1:37f.).

Zweitens, »Quantoren des Prädikatenkalküls«: Die Verwendung logischer Quantoren führt dazu, dass eine Datenbankabfrage ein Ergebnis nach der Wahr/Falsch-Logik ergibt. Zu den Quantoren gehören Aussagen wie »es gibt ein X, so das gilt ...« (Existenz) oder »für alle X gilt ...« (Allquantor). Aus den Quantoren des Prädikatenkalküls entwickelte Codd u.a. die relationalen Operatoren.

Mathemathikhistorisch betrachtet greift Codd auf die Erkenntnisse der modernen Logik zu, die auf die Arbeiten Freges und Cantors zurück gehen. In *Begriffsschrift — eine der arithmetischen nachgebildete Formelsprache des reinen Denkens* (1879) entwickelte Frege einen Propositionskalkül, welcher die Beweistheorie von Leibniz mit der Macht logischer Konnektoren verband. In dem Ansinnen, Sprache logisch verfügbar zu machen, widmete sich Frege der Bedeutung von Quantoren, wie »alle«, »einige«, »viele« oder »die meisten«. Seinen Calculus reduzierte er auf zwei Quantoren: »alle« und »mindestens

<sup>78</sup> Codd führt als Referenz für den mathematischen Teil seines relationalen Kalküls das Lehrbuch *An Introduction to Mathematical Logic* von Alonzo Church an, dessen erste Ausgabe 1944 erschien, während David Childs seine Erweiterte Mengentheorie aus der Axiomatischen Mengenlehre mit Verweis auf Patrick Suppes Veröffentlichung *Axiomatic Set Theory*, Princeton, 1960 aufbaute.

einer«, mit deren Hilfe er Propositionen (Subjekt-Prädikat Konstellationen) auf ihren Wahrheitsgehalt hin untersuchen konnte. Um diese Untersuchungen zu leisten, waren logische Konnektoren wie ›wenn... dann...‹ von Nöten. Frege erkannte, wie diese komplexeren Konnektoren in einfache Ausdrücke umgewandelt werden könnten, die ›nicht‹, ›und‹, ›oder‹ verwendeten, ohne die Komplexität der Ursprungsaussage zu beschädigen. ›Wenn dies ein Stuhl ist, dann hat er Beine‹, wurde umwandelbar in ›Es kann nicht ein Stuhl sein und nicht Beine haben‹.<sup>79</sup> Dies erlaubte den Ausbau der Figuration zur Verwendung mehrstelliger Prädikate, die über Beziehungen verknüpft sind – Relationen.

Es führt an dieser Stelle zu weit, die zahlreichen Komplikationen und Paradoxien, die in der Folge Freges Theorie entdeckt wurden, nachzuzeichnen. Festzuhalten bleibt, dass die von ihm entwickelten Quantoren und das Prädikatenkalkül, fortgeführt durch Bertrand Russel und Alfred N. Whitehead in *Principia Mathematica* (1910), in der relationalen Algebra ihre Anwendung finden.

Drittens, ›Relationale Operatoren‹: Neben den üblichen Operatoren der klassischen Mengentheorie entwickelte Codd die Operationen ›Projektion‹, ›Restriktion/Selektion‹, ›Join‹ und ›Tie‹. Mit dieser Schwerpunktsetzung legte er das Gewicht nicht auf alle verfügbaren Operationen der Klassischen Mengentheorie, sondern auf jene, die es erlaubten, neue Relationen aus Relationen zu erzeugen (Codd 1970, 385). Von diesen vier Operationen sollen jene drei, die heute maßgeblich sind, näher beschrieben werden: Erstens, ›Projektion‹: Die Projektion wählt bestimmte Domänen (Spalten) einer Relation anhand ihre Namen aus. Sie leitet sich aus der Existenzaussage der Prädikatenlogik  $\exists(x)$  ab (Church 1956, 1:42). Codd verändert die Bedeutung der Projektion in der Relationalen Algebra gegenüber der Klassischen Mengentheorie: Dort bezog sich die Projektion auf geordnete Elemente einer Menge: man spricht in der Klassischen Mengentheorie vom 1. Element, vom 2. Element und so weiter. Codd's Relationen hingegen bestehen aus ungeordneten Elemente, d.h. die Ordnung der Spalten (von links nach rechts, bzw. von 1 bis unendlich) ist irrelevant. Ausgewählt wird anhand der Namen der Spalten, nicht anhand ihrer Anordnung. Da die Domänen (Spalten) durch ihren Namen Auskunft über die Art der enthaltenen Information geben, kann die Art der zu suchenden Information bestimmt werden (Abb. 12).

supplier	part	project	quantity
1	2	5	17
1	3	5	23
2	3	7	9
2	7	5	4
4	1	1	12

FIGURE 1: A Relation of Degree 4

project	supplier
5	1
5	2
1	4
7	2

Figure 3: A Permuted Projection of the Relation in Figure 1

Abb. 12: Die Relation ›ship‹ wurde durch eine Projektion so reduziert, dass die Domänen part und quantity eliminiert wurden und eine Zeile, die doppelt vorkam (project ›5‹, supplier ›1‹) wurde auf ein Vorkommen reduziert (Codd 1969, 6).

Codd 1969/1970 setzte sich hier von David Childs vorhergehendem Ansatz ab, der mit der Operation ›i-th domain of‹ im Rahmen der klassischen Mengentheorie verbleibt. Die Domänen/Spalten können bei Childs nicht über Namen ausgewählt werden, sondern werden über ihre Ordnungszahl innerhalb der Menge angesprochen, also 1, 2, 3 ... (Childs 1970, D-20).

<sup>79</sup> Zu den diagrammatische Wendungen Freges' siehe einführend *Gottlob Frege – Eine Einführung in sein Werk* (Kutschera 1989) und zu Freges mathematikhistorischer Einordnung siehe *The Search for mathematical Roots, 1870-1940 – Logics, Set Theories and the Foundations of Mathematics from Cantor through Russell to Gödel* (Grattan-Guinness/Bos (Hrsg.) 2000, 232f.).

In der später erfolgten Implementation durch SQL entspricht die Projektion der Abfrage ›SELECT columns FROM relation‹.

Zweitens, ›Restriktion/Selektion‹: Sie ermöglicht eine Auswahl von Tupeln (Zeilen) durch eine Selektionsbedingung. Die Bedingung kann mit anderen logischen Operatoren kombiniert werden, z.B. AND, OR, NOT. Codd leitete die Logik der Restriktion her, gab jedoch 1970 kein Beispiel für ihre Anwendung, aus der die Möglichkeiten ersichtlich geworden wären. Er veränderte die Bedeutung der Restriktion in der Relationalen Algebra gegenüber der Klassischen Mengentheorie: Dort bezieht sich die Restriktion auf geordnete Elemente einer Menge, man spricht dort vom ersten Tupel, vom zweiten Tupel und so weiter. Codd's Relationen hingegen bestehen aus ungeordneten Elementen und ausgewählt wird anhand von Werten, die ein Tupel enthält, üblicherweise der Primär- oder Fremdschlüssel.

Die Zeilen werden nicht unmittelbar angesprochen (so wie die Spalten durch ihren Namen), sondern mittelbar, indem eine Angabe eines Wertes, welchen die Nutzer kennen, folgt, z.B. ›Die Farbe ist rot‹ oder ›Die Höhe liegt zwischen 12 und 16 Meter‹.

Childs demonstrierte Restriktion zweier Mengen von Tupeln aufeinander mithilfe der Operation ›Restriction of A to B‹ (Childs 1970, D-27). In der Implementation durch SQL entspricht die Restriktion/Selektion der Abfrage ›SELECT projection FROM relation WHERE selection < 200‹.

Drittens, ›Join‹. Dieser erlaubt das Verknüpfen verschiedener Relationen zu einer neuen, übergreifenden Relation anhand eines Attributes, welches beide Relationen gemeinsam haben. Ergebnis sind die Kombinationen all jener Tupel/Zeilen, deren gemeinsames Attribut zweier verschiedener Relationen gleich ist. Dies entspricht den nacheinander ausgeführten Operationen ›Kartesisches Produkt‹ (Kombination aller mit allen) und ›Restriktion‹. Auch hier ist eine Form von Mikro-Prozeduralität angelegt. Die Operationen ›Verknüpfe alles mit allem‹ UND ›Schränke das Ergebnis ein, sodass nur jene Tupel/Zeilen sichtbar sind, die über ein gemeinsamen Wert (z.B. ein Schlüssel) verfügen‹ werden nacheinander und zusammenhängend ausgeführt, um den Join zu erzeugen. Join erweitert tendenziell die Informationsmenge, indem zwei vorhandene Mengen kombiniert werden, Codd bezeichnet dieses Potenzial als ›Pluralität der Joins‹ (Codd 1969, 7).<sup>lxvii</sup>

R ( supplier part )		S ( part project )	
1	1	1	1
2	1	1	2
2	2	2	1

Figure 4: Two Joinable Relations

R*S ( supplier part project )		
1	1	1
1	1	2
2	1	1
2	1	2
2	2	1

Figure 5: The Natural Join of R with S (from Figure 4)

Abb. 13: Die Relation R und S (Fig. 4) werden mit einem natürlichen Join verbunden (Fig. 5), sodass anhand der Domäne *part* jede Zeile in R mit jeder Zeile in S kombiniert wird (Codd 1969, 6).

Joins unter Verwendung zweier Relationen ermöglichen die Darstellungen von One-to-One-Beziehungen und One-to-Many-Beziehungen. Joins mit mindestens drei Relationen erlauben Many-to-Many Beziehungen (Abb. 13). Auf diese Weise war ein Problem gelöst, dass im Netzwerkmodell häufig

auftrat: Wie konnten Vielheiten einander zugeordnet werden? Wie können beispielsweise mehrere Autoren einem Buch zugeschrieben werden, und gleichzeitig jeder einzelne Autor verschiedene Bücher geschrieben haben? Während im hierarchischen Modell sich verzweigende und zirkuläre Abhängigkeiten entstanden, die navigatorisch zu durchdringen waren, reichte im relationalen Modell eine Art Perspektivwechsel, bei dem man einmal aus der Richtung der Autoren auf die per Join verknüpften Daten schaut, um zu sehen, welche Bücher sie publiziert haben und einmal aus der Sicht eines Buches auf die per Join verbundenen Daten, um abzufragen, welche Autoren an einem Buch beteiligt waren. Diese verschiedenen Sichtweisen innerhalb des gleichen Modells, ohne Eltern-Knoten – Kind-Knoten, erlaubte das deutlich flexiblere Einfügen weiterer Attribute oder Tupel/Zeilen, ohne dass Daten komplex umkopiert werden mussten.<sup>80</sup>

In Childs' Extended Set Theorie entsprach dies der Operation ›xpan‹ (Expand). Die SQL Notation des Joins kann beispielhaft angegeben werden als: ›SELECT projection FROM relation1 INNER JOIN relation2 ON relation1.attribute = relation2.attribute‹.<sup>81</sup>

Deutlich wird, dass diese Operationen nicht dazu dienen, Bedeutung zu erkennen oder Sinn zuzuschreiben, sondern Referenz herzustellen. Die Referenz des Gespeicherten ist im Datenbank-Modell, in den Namen der Relationen und der Felder aufgehoben, in denen jeweils Daten in Formation gebracht werden, sowie in den Suchanfragen selbst. Die Kriterien zur Informationsabfrage sind mit einer ›Select-From-Where‹ Abfrage vollständig und eindeutig formulierbar (Burkhardt 2015, 259).

Diese Operationalität relationaler Datenbanksysteme, angesiedelt in der Zwischenwelt zwischen der Oberfläche des Nutzerzugriffes und den maschinellen Tiefen, auf die das Betriebssystem zugreift, unterlag einem jahrzehntelangem Entwicklungsprozess. Dieser changiert zwischen den in Publikationen formulierten theoretischen Ansätzen und den praktischen Umsetzungen durch Softwareprototypen und tatsächlichen Produktivsoftware und ist daher nur partiell zu rekonstruieren. Einige Hinweise erlauben uns die Stellungnahmen der Protagonisten der 1970er und 1980er Jahre, die zum Teil ihre eigene Beteiligung kritisch diskutieren. So weist in Bezug auf die relationale Operationalität der Experte für relationale Datenbanksysteme David McGoveran darauf hin, dass Codd's 1969 definierte relationale Algebra sich durchaus von den heute verfügbaren Operationen unterscheidet. Die Definition dessen, was eine Relation ist, und die verfügbaren Operationen, haben sich weiter entwickelt. An dieser Entwicklung war Codd mit fortwährenden Diskussionsbeiträgen beteiligt. Es gab aber auch zahlreiche Impulse anderer Entwickler von Prototypen und Datenbankanwendungen. Während der ursprüngliche Vorschlag von 1969/70 vor allem auf eine mathematisch präzise Formulierung abzielte, welche eine

---

<sup>80</sup> Heute hingegen wird das relationale Modell selbst als rigide wahrgenommen, was wiederum zu Entwicklungen wie No-SQL, objektorientierten, oder schemafreien Datenbanken (z.B. Mongo-DB) einerseits und neuen Suchmaschinentechnologien andererseits geführt hat. Dies hängt auch damit zusammen, dass Join-Operationen nach wie vor ressourcen-intensiv sind.

<sup>81</sup> Aus der klassischen Mengentheorie waren eine Reihe weiterer Operationen bekannt, die mit Hilfe der Codd'schen relationalen Algebra umstandslos und unverändert angewendet werden konnten. Das Kartesische Produkt (Kreuzprodukt) stellt die Menge aller Kombinationen der Tupel zweier Mengen dar. Jede Zeile der einen Tabelle wird mit jeder Zeile der anderen Tabelle kombiniert. Die beiden Schemata der Ausgangsrelationen werden zu einem neuen Schema vereinigt und die kombinierten Tupel ergeben eine neue Relation, deren Spezifik ist dass sie ›Alles‹ zeigt. Vereinigung/Union: Zwei Relationen, deren Schemata (d.h. die Namen und Domänen der Spalten) identisch sind, können miteinander vereint werden. So können beispielsweise einer Menge neue Elemente hinzugefügt werden. Die Vereinigung der Relationen R {A,B} und S {C,D} ergibt T {A, B, C, D}. Schnittmenge/Intersection: Die Schnittmenge zweier Relationen wird dargestellt, wobei die Schemata beider Relationen übereinstimmen müssen. Die Schnittmenge R {A, B} und S {A, C} ergibt T {A}. Ausschluss/Difference: Die Tupel (Zeilen) einer Relation R abzüglich der Tupel (Zeilen) einer anderen Relation S. Voraussetzung ist wie in Union und Intersection dass beide Tabellenschemata übereinstimmen. Die Differenz von R {A,B} und S {A,C} ergibt T {B, C}.

Integrität des relationalen Calculus anstrebte, erforderte der informatorische Alltagsgebrauch eine Ausweitung der Möglichkeiten zur Datenrepräsentation (McGoveran 2005, 1). Diese Spannungen führten so weit, dass Codd/Date vermerkten, SQL sei bis zum verabschiedeten Standard SQL-92 im Sinne der Aufsätze von Codd 1969 und 1970 nicht einmal relational gewesen. Sie hätten sich damals aber mit der Situation arrangiert, damit sich der relationale Ansatz überhaupt durchsetzen konnte.

#### 2.1.5.4 Zusammenfassung

Die Erfassung von Informationen in der relationalen Datenbanklogik erzeugt eine Informationsfläche, mit distinkten, voneinander abgetrennten flachen Mengen. Diese Mengen existieren nebeneinander, jede für sich erweiterbar, erneuerbar und entfernbar. Abfragen mithilfe einer deklarativen Query-Sprache erzeugen zwischen diesen disparaten Mengen temporäre, für die Laufzeit der Abfrage gültige Teilmengen, Überschneidungen, Vereinigungen, Gleichheiten, Differenzen oder Komplementäre, Produkte. Diese Ein- und Ausschlüsse verhalten sich den zugrundeliegenden Informationen gegenüber konstruktiv, denn sie erzeugen informatorisch Neues. Die in der Fläche nebeneinander verteilten Informationsmengen unterliegen dabei keiner Hierarchie der Nähe und Entfernung, d.h. selbst die am weitesten voneinander entfernt befindlichen Informationsmengen können über Joins miteinander in Verbindung gebracht werden. Wirksam wird hier allein die Ökonomie der verfügbaren Rechenleistung beim Abgleich großer Informationsmengen. Aufgrund der Closed-World-Assumption und der propositionalen Logik von Wahr und Falsch, wird Information auch in ihrer Negation wirksam. Der Ausschluss ist ebenso wirksam, wie der Einschluss.<sup>82</sup>

Der Umstand, dass die relationale Algebra in der Mengenlehre begründet ist, positioniert die Konstruktion von Informationen mittel Abfrage als Modi der Distribution, der Dispersion und der Assoziation. Während die Distribution den Tribus, den Stamm oder Bezirk anspricht, und die Dispersion den Part, den Teil adressiert, verweist Assoziation auf den Sozius, den Gefährten. Die Modi der relationalen Algebra, erlauben den Zugriff auf Information ausgehend von Bezirken, einander abgegrenzten Bereichen. Information als Formation wird geteilt und verteilt, und sie wird im Rahmen mathematischer Logik, erneut zusammen gesetzt und neu assoziiert. Bedeutung wird qua Referenz durch Namen erzeugt.

#### 2.1.6 Fazit

Erstens, ›Mannigfaltige Genealogien‹: Es konnte gezeigt werden, auf welche zahlreichen verschiedenen Ideenstränge die durch Codd formulierte relationale Algebra zurück verweist. Levien/Maron diskutierten eine relationalen Informationssprache, welche im Grunde eine auf dem relationales Kalkül aufbauende Abfragesprache ist. Die wichtige Idee daran war, dass relationale Algebra ein abgeschlossenes Modell (finite model) erzeugen können, auf welches mit einer logischen, nicht-prozeduralen Abfragesprache zugegriffen werden kann. Davies warf die Notwendigkeit der Datenunabhängigkeit auf und demonstrierte, wie dies über Schlüssel realisierbar ist. Childs präsentierte eine Theorie von Datenrelationen zusammen mit einer mengentheoretischen Abfragesprache, die als relational verstanden werden

---

<sup>82</sup> Dies führte zum Beispiel beim Bibliotheksinformationssystem OPAC dazu, dass man sehr genau wusste, wenn es ein Buch in einer spezifischen Bibliothek nicht gab. Es war dann nicht verzeichnet. Allerdings konnten die Nutzer nicht überprüfen, ob das gesuchte Buch überhaupt existiert, da in der Closed-World-Assumption nur existierte, was verzeichnet ist.

kann. Bei Mealy ist im Zuge der Erstellung einer Familiengenealogie der Begriff der Relation prominent. Codd gelang es, diese Ansätze zu synthetisieren und zu zeigen, dass relationale Algebra und relationaler Calculus miteinander vereinbar waren.

Auffallend an den Konstellationen in der Entwicklung des relationalen Konzepts ist, wie sehr es im mathematischen Denken bzw. einer algebraischen Logik verwurzelt ist. Seine Befürworter finden sich unter Logikern und Mathematikern, wie Roger Levien, Melvin Maron, David Childs, Edgar Codd, Chris Date, Ed Oates, Bob Miner, Larry Ellison und Michael Stonebreaker.<sup>83</sup> Im Gegensatz dazu sammeln sich unter den Entwicklern des hierarchischen und des Netzwerkmodells (CODASYL) maschinenorientierte Positionen: Bachmann war der Ausbildung nach Ingenieur. Es deutet sich in den Auffassungen zur Datenorganisation und dem Erfolg der relationalen Logik als hardware-unabhängige Konfiguration in den Datenbankfigurationen ein genereller Paradigmenwechsel an, jener von der Hardware-orientierten Entwicklung des Computings zur Algorithmen-getriebenen Entwicklung. Mit Haigh (2006) ist zu vermerken, dass Bachmann die Formulierung vieler konzeptueller Ansprüche an Datenbanken in den 1960er Jahren zu verdanken ist – worauf der relationale Ansatz in der Folge aufbaute.

Zweitens, ›von der Maschinenteknik zur Softwaretechnik‹: Die Einführung des relationalen Datenmodells markiert den softwareseitigen Nachvollzug des Umbruchs von sequentiellen zu direktem Datenzugriff, von der Bandmaschine zur Festplatte. Sie kennzeichnet damit auch die zunehmende Übertragung von analogen Alltagspraxen des Büros, welche durch ein menschlich-maschinelles Zusammenspiel gekennzeichnet waren, in Softwareprozesse.<sup>84</sup> Motiviert sind beide Entwicklungen durch die Grenzen des Konzepts Batch-Processing, bei dem ein Stapel von Aufgaben eines singulären Nutzers von Anfang bis Ende durchgearbeitet wird.

In der neuen Softwarefiguration ›Datenbankmanagementsystem‹ sind nicht mehr die Nutzer für die Speicherverwaltung zuständig, sondern diese wird durch den Computer übernommen. Automatisierbare kooperative Praxen wurden somit der prozessierenden Symbolverarbeitungsmaschine übertragen. Damit erweitern sich die möglichen Gebrauchskontexte und die potentielle Nutzerbasis, denn das Wissen um die Datenstrukturen und die Adressierung im maschinellen Kontext war nicht länger Nutzeraufgabe. Zudem wurde der Umstieg zwischen verschiedenen Hardwaregenerationen einfacher, da nur jener Teil der Software geändert werden musste, der die logische Datenstruktur auf die physikalische Speicherstruktur abbildet. In der Folge sank der Investitionsbedarf in Software bei Systemwechseln.

Wichtige Voraussetzung dafür war eine Abstraktionsleistung, welche die Materialität der Maschine verließ und sich in der Materialität der Software (der relationalen Logik der Datenorganisation) neu fand. Diese zusätzliche Schicht, die zwischen Materialität der Maschine und Logik der Informationsstruktur vermittelt, sorgt für ein Entschwinden des Indexikalischen der Datenorganisation. Ließ sich bis zur Einführung relationaler Algebra, mit dem Blick auf die Daten und auf die Struktur der Daten, Rückschlüsse ziehen auf die maschinelle Organisation und den Zustand der Maschine, so war dies fortan nicht länger möglich.

---

<sup>83</sup> Gugerli argumentiert in *The World as Database* ähnlich, indem er den Unterschied als kulturellen Kontrast zwischen Techniker und angewandtem Wissenschaftler markiert (Gugerli 2012, 298).

<sup>84</sup> Wobei die vollständige Übertragung von Prozessen in Software eine Illusion geblieben ist. Zwar wurden automatisierbare Prozesse in Software eingeschrieben, doch wird auch deren Funktionieren immer durch menschliche Interaktion sichergestellt.

Drittens, ›Tableauisierung der Information‹: Bei der relationalen Algebra handelt es sich, im Unterschied zu vorhergehenden Speichertechniken, um eine mathematisch fundierte Formalisierung. Die Information wird nicht mehr entlang der physikalischen Datenträger und ihrer maschinellen Konfiguration formiert, sondern entlang einer abstrahierten Logik, die der Logik menschlichen Denkens nahe kommen sollte. Diese mathematische Formalisierung ermöglicht nun die Theorie-geleitete Entwicklung von Abfragesprachen und macht diese mit Mitteln der Logik verifizierbar.

Das relationale Datenmodell verschiebt nicht nur den Fokus von den Programmierern auf die Alltagsnutzung. Es strebt auch eine Entprozessualisierung des Informationszugriffes an. Die Verwendung einer deklarativen Abfrage um ›flache‹ Informationsmengen zu erschließen, sorgt für eine Tableauisierung von Informationen. Diese geht einher mit der Enthierarchisierung der Informationsstrukturen. Bildlich gesprochen entsteht so ein prozedurales Voranschreiten in der Informationsfläche. Abfragen können durch die Kombination mehrerer Relationen die Informationsfläche erweitern und neu figurieren.

Viertens, ›Abfragen als epistemisches Werkzeug‹: Für die relationale Datenbank wird die Closed-World-Assumption einer abgeschlossenen und endlichen Datensammlung wirksam. Relationale Datenbanken operationalisieren Wissen, indem strukturierte Informationen als Mengen aufgezeichnet werden, die miteinander überlagert oder kombiniert und gegeneinander ausgeschlossen werden können, und die durch Updates veränderbar sind. Sie werden im Zuge der Abfrage mit relationaler Algebra jeweils neu kombiniert und zwar entlang einer Prädikatenlogik, die als Werte nur ›True‹ und ›False‹ kennt.

Es ist der Ad-Hoc Charakter dieser Auswahlmöglichkeiten aus abgegrenzten (wenn auch nicht übersehbaren) Informationsmengen hervorzuheben, der zu einer Flexibilisierung in den Datenbank Anfragen führt. Das spezifische ist, dass aus Relationen logisch valide Relationen neu erzeugt werden und zwar in potentiell unendlicher Zahl, eine »Multiplikation von Multiplen« (Mackenzie 2010, 6). Diese Relationen-aus-Relationen-Potentialität erzeugt die Vielfalt und Flexibilität in begrenzten Informationsbeständen.

Fünftens, ›Information als Formation‹: Mit der relationalen Algebra rückt ein neuer Informationsbegriff in den Vordergrund, welcher sich von der On-Off-Logik des Signal-Noise und der Frage der Übertragung unterscheidet. Er stellt nicht mehr die Frage nach Schwellenwerten, ab denen ein Ereignis mehr ist als Noise, und nach der Verteilung von Ereignissen in einer Zeitrelationen, wie in der Aufzeichnung physikalischer Experimente. Vielmehr beschreibt ›In Formation‹ in relationalen Mengen die Formierung einzelner Informationsobjekte untereinander als bedeutungskonstituierend.

Im Anschluss an Mealy (1967, 525) ist ein Bezug von Daten auf Realität aufzurufen, welcher durch das jeweilige Informationsmodell der Softwareanwendung bzw. der Datenbank Anwendung vermittelt wird. Darin bezieht sich die Wirklichkeit alltäglicher Praxen auf die reale Welt. Ideen, welche in der menschlichen Wahrnehmung als Modell der Wirklichkeit existieren, werden mit Hilfe der relationalen Logik softwareseitig und prozessierbar modelliert. Im Speicher werden Symbole abgelegt, welche ihre Referenz dadurch erhalten, dass sie im Informationsmodell mit Namen versehen wurden. Bedeutung wird in diesem Modus nicht erkannt, wie es die Hoffnung der kybernetisch inspirierten Künstlichen Intelligenz der 1950er und 1960er Jahre war, sondern als Referenz zugeschrieben. In den Namen, welche die Domänen bezeichnen, ist jene Semantik, welche nur Menschen zu verstehen und zu erzeugen in der Lage sind, und welche maschinelle Figurationen nicht im menschlichen Maße begreifen können, realisiert. Dies ist Voraussetzung für die automatisierte Verarbeitung von Informationen durch Computer.



Erst der Umgang mit Namen und Mengen (Relationen) erlaubte eine Sicht auf Daten, welche sie durch Algebra, also durch Rechenoperationen überhaupt verarbeitbar macht. Die relationale Algebra ist in der Lage, alle möglichen Kombinationen von Daten zu erzeugen, da es sich um Rechenoperationen wie Addition, Subtraktion, Multiplikation usw. handelt, kombiniert mit den Booleschen Operatoren wie UND, ODER, NICHT.

Wie wird nun zwischen diesen automatisierten Abstraktionen und den human-maschinellen Praxen vermittelt? Die Verwendung der Tabelle bietet als Metapher für die Menge ein vertrautes visuelles Interface und trug zum Erfolg des relationalen Konzepts bei. Ihr ist daher im Folgenden Abschnitt besondere Aufmerksamkeit gewidmet.

## 2.2 Operationalisierung – Tabellieren

### 2.2.1. Tabellarische Materialität und Operationalität

»Es ist eine Matrix aus vertikalen Spalten und horizontalen Zeilen« (Goody 1977).<sup>lxxviii</sup>

Tabellen werden zwar ›angesehen‹, ›gelesen‹, ›überflogen‹, der ›Nachschau unterzogen‹ oder ›berechnet‹, nie aber werden sie ›tabelliert‹ oder ›tabuliert‹ – letztere Verben sind heutzutage außer Gebrauch geraten. Der Hinweis darauf, dass einst tabuliert wurde, dieses Wort inzwischen aber fast vergessen ist, ruft die Historizität tabellarischer Praktiken auf. Es deutet sich an, dass es spezifische Materialitäten und Praxen des Tabellierens gibt, welche historisch unterschiedliche Ausformungen erlebten. Diese Aufzählung möglicher Operationen mit Tabellen zeigt darüber hinaus die praxeologische Orientierung an, welche den folgenden Abschnitt durchzieht. Die Tabelle speichert nicht nur, sie erlaubt und erfordert es auch, zu registrieren und zu identifizieren. Sie überträgt nicht nur Informationen zwischen verschiedenen Kontexten, sondern erlaubt, wie zu zeigen sein wird, das Delegieren zwischen menschlichen und nichtmenschlichen Agenten und das Fortschreiben von Operationsketten. Sie wird nicht nur bearbeitet, sondern setzt selbst Arbeit in Gang und dient damit der Koordination sozialer Verhältnisse (vgl. Gießmann 2018, 108). Dieser praxeologischen Wendung ist das vorliegende Kapitel gewidmet. Tabellen sind nicht nur in der tiefen Humangeschichte eines der wichtigsten epistemischen Werkzeuge der Menschheit. Sie haben in der jüngeren Geschichte als Hybrid aus Textualität und Visualität eine wichtige Rolle in der Durchsetzung des relationalen Datenmodells gespielt. Erst als die mathematischen Strukturen der Menge, in denen sich die relationale Algebra entfaltet, mit der visuellen Metapher der Tabelle bekleidet wurden, nahm das Verständnis des relationalen Modells jenseits enger Mathematikerkreise zu. Angesichts dessen gilt es, die operationale Bedeutung des Tabellierens im kulturtechnischen Vollzug herauszuarbeiten.<sup>85</sup> Diesem Anliegen widmet sich das vorliegende Kapitel in einem Dreischritt. Es untersucht

- tabellarische Materialitäten und Operationalitäten,
- verschiedene historische tabellarische Praktiken und
- Übergänge tabellarischer Praktiken und Operationalitäten in Datenbankpraktiken.

---

<sup>85</sup> In der einschlägigen medientheoretischen Studie zur elektronischen Datenbank von Markus Burkhardt und in den medienhistorischen Studien zur Emergenz der (relationalen) Datenbank von Thomas Haigh wird die Tabelle *én passant* erwähnt, ohne intensiver auf ihre spezifische Materialität und Operationalität einzugehen. Allein der Aufsatz von Markus Krajewski *In Formation – Aufstieg und Fall der Tabelle als Paradigma der Datenverarbeitung* (Krajewski 2007) geht intensiv auf die Bedeutung der Tabelle für die relationale Algebra ein. Latour, vermerkt zu Inskriptionen, zu denen auch die Tabelle zu zählen ist: »Die Praktiken, an denen ich interessiert bin, wären sinnlos, wenn sie nicht auf bestimmte Kontroversen Einfluss hätten und Kritiker dazu bringen würden, neue Fakten zu glauben und sich auf neue Art zu verhalten. Hier versagt ein ausschließliches Interesse an Visualisierung und Schrift und kann sogar kontraproduktiv sein« (Latour 2006, 266). Damit seien Inskriptionen nicht auf ihre Visualisierung hin allein zu befragen, sondern in Bezug auf deren Mobilisierung. »Kurz: Man muss Objekte erfinden, die mobil, aber auch unveränderlich, präsentierbar, lesbar und miteinander kombinierbar sind« – »unveränderlich mobile Elemente« (ebd., 266f.). Er gibt damit einen deutlichen Hinweis darauf, dass Tabellen nicht allein visuelle Argumente sind, sondern auch mehr sein können, und zwar, wie gezeigt werden soll: operationale Denkräume.

Im akademischen Diskurs ist das Tabellieren, wenn nicht gänzlich ignoriert, vorrangig als Visualisierung und als Wissensspeicher verstanden worden, als diskursiver Ursprungsakt in dessen Zuge Wissen konstituiert wurde und Macht über die geordneten Dinge entsteht. Es könnte leicht der Eindruck entstehen, die Tabelle diene allein der Visualisierung von Gewusstem. Doch werden Tabellen nicht allein aufgestellt und in diesem Zuge geordnet und gruppiert, um ein finales Gebilde zu ergeben. Sie sind vielmehr, wie in diesem Kapitel aufzuzeigen ist, operational eingebunden in komplexe Wissens- und Prozessfigurationen.

Ziel dieses Kapitels ist es, die Tabelle nicht allein als Mittel der Visualisierung zu positionieren. Es wird vielmehr ihr Werkzeugcharakter herausgearbeitet, die Modi der Operationalisierung, ihre wissensproduzierenden Qualitäten und ihre Funktion im Sinne der Kulturtechnikforschung.

Dies erfordert verschiedene methodische Zugänge. Dabei wird das Argument der Visualisierung nicht einfach über Bord geworfen, vielmehr werden Fragen der Visualisierung bzw. der Diagrammatik zum Ausgangspunkt genommen, um die Operationalisierung zu beschreiben. Die visuellen Qualitäten von Tabellen verstehen sich als ›Materialität‹, welche die möglichen Operationalisierungen mitformt. So diskutiert der Abschnitt um Konstruktion und Informationstopologien das stets implizite Verhältnis zwischen den Daten und der Struktur und Topologie der Tabelle. Mithilfe von Erkundungen der Diagrammatik werden Raum- und Zeitordnungen identifiziert, welche die Tabelle überhaupt erst lesbar machen. Die Lesbarkeit wiederum ist Voraussetzung für Wissensoperationen, wie das Füllen leerer Zellen, das Lesen von Zellen und Zusammensetzen von Informationen im Sinne einer Referenzialität, sowie das Changieren zwischen Detail und Übersicht, die als Set epistemischer Techniken zu beschreiben sind.

Diesen Ausführungen folgen vier Felder der Tabellenpraktiken, dargestellt anhand von historischen Fallbeispielen. Der Diskussion von Tabellen zur Ordnung, Speicherung und Memorierung von Wissen schließen sich Tabellen an, welche Statistik, Organisation und Prozesslogik abbilden, darauf Transaktionspraktiken mit Tabellen und abschließend mathematische Tabellen, welche maßgebend für die Publikation rechnerischer Ergebnisse wurden. Obgleich die Darstellung auf historische Beispiele zurückgreift, geht es nicht darum, historische Kontinuitäten zu erzeugen, dafür erfolgen viel zu große zeitliche Sprünge, deren Zwischenräume unerkundet bleiben. Vielmehr spielen die Frage nach der Operationalität der Wissenserzeugung und die Zugänglichkeit des Materials eine wesentliche Rolle in der Auswahl.

Zahlreiche Autoren haben sich mit Einzelaspekten des Tabellierens auseinander gesetzt und das Material ist disparat über verschiedene, sich teils überschneidende Wissensfelder verstreut: Historiographie,<sup>86</sup> Anthropologie,<sup>87</sup>

---

<sup>86</sup> Die Monographie von Benjamin Steiner unternimmt eine präzise Historiographie der (Wissens-)Tabelle in der frühen Neuzeit (Steiner/Benjamin 2008). Ein von Martin Campbell-Kelly et. al. herausgegebener Sammelband widmet sich der Geschichte mathematischer Tabellen (Campbell-Kelly u. a. (Hrsg.) 2003). Verschiedene Enzyklopädien wurden danach befragt, wie sich der Begriff der Tabelle zeitlich darin widerspiegelt (Zedler 1744; Glaisher 1911). Auf David Griers Monographie über Human Computer konnte zurückgegriffen werden, um die Erstellung mathematischer Tabellen in der historischen Perspektive der vor-elektronische Rechenära aber auch im Hinblick auf deren Operationalität zu erkunden (Grier 2003; Grier 2005). Kammer, Kasten und Tafel untersuchte Barbara Segelken in einer Monographie als »Bilder des Staates« in der Periode der Herausbildung des modernen Staatswesens (Segelken 2010).

<sup>87</sup> Jack Goody diskutierte anhand der Liste die Bedeutung diagrammatischer Wissensorganisation als anthropologische Konstante und unterzog sie in einem Kapitel seiner Monographie erstmals einer (Selbst)-Kritik als Wissenswerkzeug der Anthropologie (Goody 1977). Cornelia Vismann kennzeichnete die Liste im Verhältnis zur Datei als eine prozessuale Abfolge von Schritten (Vismann 2008, 7–9).

Diagrammatik und Philosophie der Ästhetik,<sup>88</sup> Kommunikationsdesign,<sup>89</sup> Soziologie<sup>90</sup> und der Medientheorie.<sup>91</sup> Ohne Anspruch auf Konsistenz oder gar Synthese, sondern mit dem bereits erwähnten Fokus der Operationalität der Wissenserzeugung, dient das folgende Kapitel der Zusammenführung dieser verstreuten Erkundungen. Es gilt, einen Überblick zu verschaffen, über das Denkwerkzeug, welches Überblick verschafft.

Diese Untersuchung geht einher mit der Leerstelle einer Wissenspraxis, die mit der Tabelle engstens verbunden ist: dem Formular. Nicht allein aus Zeit- und Platzgründen wird in der vorliegenden Studie die eine dem anderen vorgezogen. Während das Formular dem Eintragen des einzelnen Datensatzes dient, erlaubt die Tabelle sowohl die Sicht auf den einzelnen Datensatz als auch den Überblick über die Gesamtheit der Datensätze. Täglich treffen wir auf den Benutzeroberflächen auf das simple Formular, das derart oft gesehen, unerkannt bleiben: das Suchfeld. Da jedoch die in der relationalen Algebra operationalisierte Mengentheorie ihren Ausdruck in der Tabelle fand, wird dieser hier der Vorzug gegeben und die Leerstelle des Formulars riskiert.

Eine weitere Einschränkung soll erwähnt werden: Ausgehend vom Focus auf Datenbankpraxen überspringt die folgende Diskussion das Eingebundensein von Tabellen in einen übergeordneten Diskurs, z.B. das Verhältnis zu einem umgebenden Fließtext, oder das Verhältnis zu einem Argument eines Autors. Ebenso wenig wird die Frage verfolgt, wie die Tabelle als Inskription in das Herstellen von Objektivität und Wissenschaftlichkeit eingebunden ist. Diese Einschränkungen der Untersuchung haben zeitpragmatische Gründe.<sup>92</sup>

Die Materialität der Tabelle bedingt ihre Operationalität – ein Beitrag zur Materialität der Tabelle ist somit ein Beitrag zur Materialität der Softwarefiguration Relationales Datenbankmanagementsystem (RDBMS). Ziel ist es, die Voraussetzungen herauszuarbeiten, welche dazu geführt haben, dass die

---

<sup>88</sup> Markus Krajewski hat eindringlich auf die Wichtigkeit der Tabelle als räumlich organisiertes Paradigma der Datenverarbeitung hingewiesen. Er etablierte einen neuen Begriff der Information, welcher darauf hinweist, dass Wissen in Formation gebracht wurde (Krajewski 2007). Die Einzelbeiträge von Sybille Krämer diskutieren ästhetische Dimensionen der Linie und Graphur gemeinsam mit Räumlichkeit als wesentlichem epistemologischen Mittel am Übergang zur Diagrammatik als Form visuellen Denkens (Krämer 2009; Krämer 2010a; Krämer 2010b; Krämer 2011). Dieter Mersch diskutierte in einem Beitrag visuelle Argumente zwischen Zeugenschaft und Konstruktion (Mersch 2006), während Horst Bredekamp, Birgit Schneider und Vera Duenkel mit dem Begriff des »technischen Bildes« das Feld der Bildwissenschaften im deutschsprachigen akademischen Diskurs verankerten (Bredekamp/Schneider/Duenkel (Hrsg.) 2008).

<sup>89</sup> In der visuellen Kommunikation wird die optimale Darstellung quantitativer Informationen von Edward Tufte, und im Anschluss daran von Stephen Few diskutiert. Zu den Mitteln der Wahl zählen die Autoren neben Diagrammen und Infografiken auch die Tabelle (Tufte 1983; Tufte 1995; Tufte 1997; Few 2012).

<sup>90</sup> Zur Bedeutung von Statistiken für die moderne Subjektkonstitution und Ordnungen des Wissens siehe *Archäologie des Wissens* (Foucault 1969; Foucault 1978).

<sup>91</sup> Gloria Meynen kennzeichnet das Büro in Hinsicht auf Materialität und Operativität der Bild- und Schreibflächen in Hinsicht darauf, welche Materialitäten das Verarbeiten von Informationen und das Rechnen im Büro nehmen (Meynen 2004). Cornelia Vismann beleuchtete die Materialität von Akten und Registratur als Ordnungssystem des juristischen Wissens, welche nicht allein individuelle Rechtssprechung, sondern auch die staatliche Rechts-Konstitution formatierten (Vismann 2008). Markus Krajewski widmete sich einer Monographie 2008 ebenfalls den Materialitäten des Bürokratischen, indem er Zettel, Karteikarten, Karteikästen als Ordnungssysteme des Wissens untersuchte (Krajewski 2011). Ein Sammelband von 2016 fasst schließlich verschiedenste Materialitäten und Praktiken als *Medien der Bürokratie* (Balke/Vogl/Siegert (Hrsg.) 2016).

<sup>92</sup> Folgte man der Argumentation Latours, so würde die Tabelle, die bei ihm unter den Begriff »Inskriptionen« fällt, außerdem zum Argumentationswerkzeug und zwar im Zuge des »Staging einer Szenographie in der die Aufmerksamkeit auf einen bestimmten Satz dramatisierter Inskriptionen konzentriert wird« (Latour 2006, 283). Mit Latour würde man verstärkt danach fragen, auf welche Weise mithilfe der Tabelle »jemand einen anderen davon überzeugt, eine Aussage aufzunehmen, sie weiterzugeben [...]« (ebd. 264).

Tabelle in der Operationalität relationaler Datenbankmanagementsysteme, eine unterschätzte und bei genauerer Betrachtung äußerst grundlegende Bedeutung erfährt.

### 2.2.1.1 Tisch – Tafel – Tabelle

»die Rechnung der Planeten Läufe. [...] So hab' ich / diese beschwerlichkeit / zu erleichtern / mit großer langwieriger müh / diese Tabelln ausgerechnet / die hiermit den Lesenden in öffentlichen Druck communicir't werden« (Maria Cunitz 1650, 184).

Es gibt einen Zusammenhang zwischen der lateinischen *tabula* (dt. *Tafel*), einem Schreibbrett, zum romanischen *table* als Tisch, auf dem etwas ausgebreitet werden kann, oder *table* als Tabelle, sowie dem *tableau* als Unterbrechung einer Theaterszene, während der alle Akteure still stehen und somit eine Überschaubarkeit der Szene erzeugen.<sup>93</sup> Diese Wortverwandtschaft kommt nicht von ungefähr, denn *table* und *table* haben eines gemeinsam: die räumliche Dimension des Ausbreitens von Gegenständen. Tisch und Tabelle treffen sich ab dem 12. Jahrhundert als *table* bei den Exchequer, den Zählmeistern der normannischen und später englischen und schottischen Könige (Abb. 14). Die Abbildung *Exchequer of Ireland* zeigt drei Richter links vom Tisch. Auf diesem liegen Kerbhölzer und Zählmarken, die Geld repräsentieren. Unten sitzt ein Sheriff und rechts diejenigen, die ihren Fall vor dem Exchequer vortragen. Oben am Tisch sitzen Schreiber, die den Prozess dokumentieren.<sup>94</sup>



Abb. 14: Exchequer of Ireland. Der Tisch ist mit einem Tuch bedeckt, dessen Schachbrettmuster in Form und Funktion auf die Tabelle verweist (The National Archives 1450).

<sup>93</sup> Vgl. auch (Didi-Huberman/Charcot 1982, 24, 30).

<sup>94</sup> (The National Archives 1420); Vgl. auch (Menninger 1898, 2013:236ff.).

Richard F. Nigel beschrieb um 1180 die Vorgehensweise auf der 10×5 Fuß umfassenden Tischfläche: »Er hat rechterhand, im untersten Feld, einen Haufen Pennies ab 11 abwärts platziert; im zweiten Feld Schillinge ab 19 abwärts;<sup>95</sup> im dritten Feld jedoch Pfund. Und dieser Haufen sollte gegenüber und direkt vor ihm liegen, da in den normalen Konten des Sheriffs, dies die Mitte darstellt; im vierten Feld befindet sich ein Haufen Zwanziger [Pfund], im fünften Feld die Hunderter [Pfund], im sechsten die Tausender [Pfund]; im siebten die zehntausend Pfund, was jedoch selten vorkommt, zum Beispiel, wenn der König oder sein Beauftragter, eine Aufstellung der Einnahmen, die das ganze Königreich von den Noblen des Königreichs erhalten hat, durch den Schatzmeister und die Kämmerer erhält« (Nigel 1180).<sup>lxxxix</sup> Beim Exchequer fallen nicht nur Tabelle und Tisch ineinander, es zeigt sich auch eine grundlegende Funktion der tabellarischen Struktur: Sie stellt eine Syntax zur Verarbeitung von Informationen zur Verfügung.

Rechenhilfen, die über ihre räumliche Struktur eine diagrammatische Lösung<sup>96 lxxx</sup> von Rechenaufgaben ermöglichten, sind seit Jahrhunderten geläufig. Dazu zählen die Zähl- und Rechentische wie die Marmor-Tafel von Salamis (300 v.u.Z.) von 150×75 cm Fläche oder der Zähl Tisch aus dem 16. Jahrhundert von 153×98 cm Fläche in der Sammlung des Musee de l'Oeuvre Notre Dame in Strasbourg (Schärlig 2006, 127). Ebenfalls belegt sind ähnlich strukturierte Zähltücher, die auf Tischen oder auf dem Boden ausgebreitet wurden. Sie ermöglichten das *Rechnen auf Linien*, wie es Adam Ries im 16. Jahrhundert beschrieb.<sup>97</sup> Zu den räumlich organisierten Rechenhilfen zählen die Abakusse, die seit 2300 vor unserer Zeit, von den Sumerern und Babyloniern, circa 200 v.u.Z. aus China bis in die Jetztzeit belegt sind.<sup>98</sup> Geradezu exemplarisch für das Genre tabularischer Rechenmittel sind die Rechenstäbchen des Mathematiker John Napier (1550–1617), die tabellarisch nebeneinander gelegt Multiplikation und Division erlaubten.<sup>99</sup>

Historisch sind die Wissensoperationen des Rechnens und der Informationsverarbeitung eng mit medialen Technologien verknüpft, die sich durch räumliche Verteilung auszeichnen. Wenn in der Vergangenheit historische Medientechnologien für die Informationsverarbeitung »wiederentdeckt« wurden,<sup>100</sup> so gilt es, in diese Medientechnologien die Tabelle als Kulturtechnik nicht nur einzureihen, sondern darzustellen, dass sie vielmehr den Zählischen und -tüchern, den Abakussen, dem Rechnen auf Linien, oder den Napier-Stäbchen vorausgeht und zu Grunde liegt (Abb. 15).

---

<sup>95</sup> Zwölf Pennies entsprachen zu dieser Zeit einem Schilling; Zwanzig Schilling entsprachen einem Pfund; also 240 Pennies = 1 Pfund.

<sup>96</sup> Vgl. auch Krämers Diskussion des Calculus als symbolische Maschine (Krämer 1988).

<sup>97</sup> Die zugrundeliegenden mathematische Praktiken existierten bereits. Ries' Verdienst lag in der Sammlung und Publikation dieser Praktiken.

<sup>98</sup> Vgl. *Büro. Dissertation* (Meynen 2004, 130–139, 151f.).

<sup>99</sup> Der schottische Mathematiker Napier beschrieb 1614 erstmals das mathematische Prinzip des Logarithmus in *Mirifici Logarithmorum Canonis descriptio* und entwickelte in der Folge die »Napier sticks« als Rechenhilfe. »Mittels eines aus dem Altertum stammenden numerischen Schema, namens Arabisches Gitter [Gittermultiplikation – F.H.], ordnete er ein spezielles Set von vierseitigen Holzstäbchen an; Es gab einen Stab oder ein mit Nummern versehenes Stäbchen für jede der zehn Ziffern, einschließlich der Null. Neperianische Stäbchen oder Knochen, wie sie genannt wurden, stellten im Grunde eine Multiplikationstabelle, die in bewegliche Spalten zerteilt war, dar« (Augarten 1984, 9).

<sup>100</sup> Siehe zum Beispiel *Bit by Bit – An illustrated History of Computers* (Augarten 1984, 3–9).



Abb. 15: Neperianische Rechen-Stäblein, hier in einer flachen Ausführung. Michael Scheffelt (Hg.), Ulm, 1614 Kupferstich auf Papier auf Karton kaschiert, Letterndruck, Brokatpapier L. 7,8 cm, B. 0,8 cm (CC BY-SA 3.0 Bernd Gross).

### 2.2.1.2 Konstruktion und Information

»Und ohne mich mehr als ein paar Zentimeter zu bewegen, entfalte ich vor Ihren Augen Ziffern, Diagramme, Stiche, Texte, Umrisse und zeige hier und jetzt Dinge, die weit entfernt sind und mit denen nun eine Art von Hin- und Rückverbindung hergestellt worden ist« (Latour 2006, 276).

### AUFZÄHLUNG UND ERZÄHLUNG

Wie begründet sich der durchdringende und grundlegende Erfolg das tabellarischen Prinzips? Versuchen wir im Folgenden eine erste Annäherung: Tabellen organisieren Informationen in einem räumlichen Verhältnis zueinander. Sie dienen dazu, einen Text als Tableau zu organisieren, »dem Vor-Augen-Stellen eines Gesagten, als wäre es ein Bild« (Campe 2002, 241). Die Tabelle stelle eine Rekapitulation oder auch Aufzählung dar. Ihre Aufgabe ist das ordnende Wiederholen und Zusammendrängen des Erzählten. Versteht man die Liste als einspaltige Tabelle, lässt sich mit Jack Goody der Umstand hervorheben, dass die Liste und somit die Tabelle,<sup>101</sup> gegenüber der gesprochenen Sprache die Sortierung von Gegenständen, die Nummerierung und Gruppierung befördere. Sie verdeutliche Gruppierungen und erlaube einen höheren Grad an Abstraktion. Die Liste ist in Goodys Argumentation nicht allein eine Innovation innerhalb der Kulturtechnik ›Schrift‹, sondern die Liste gehe der kontinuierlichen Schrift

<sup>101</sup> Goody stellt die Liste in einen Zusammenhang mit der Tabellenspalte.

sogar voraus, da sie bereits, so Goody, in proto-alphabetischen Zeichensystemen der Sumerer genutzt wurde.<sup>102</sup> lxxxi

Sowohl in der von Campe auf den verschrifteten Fließtext bezogenen, als auch in der von Goody auf den Sprachfluss bezogenen Argumentation wird das Erzählte zum Aufgezählten. Der Unterschied soll mit folgendem Beispiel belegt werden: Im Fließtext unterbrechen allein Zeilenumbrüche und Absätze den linearen Lesefluss. Sie geben dem Text eine listenförmige Struktur und trennen Sachverhalte graphisch voneinander ab. Jenseits dieser Brüche erzeugt die Leserichtung des Fließtextes Kontinuität und bildet eine kausale Struktur, aus der einzelne Daten oder Sachverhalte nur schwerlich zu isolieren sind: Zu sehr ›kleben‹ die Daten in der qua Syntax determinierten Wort- und Zeichenabfolge der Sätze. Mit Edward Tufte ist zu argumentieren, dass, indem der Umbruch der Zeilen sich an der Spaltenbreite des gesetzten Textes orientiert, die im Fließtext eingeschriebenen Informationsobjekte räumlich zufällig verteilt werden und daher keine visuelle Relation zwischen ihnen entstehen kann (Tufte 1983, 178).

Sie besaßen *fünf* schwarze Katzen, *zwei* weiße und *dreizehn* gescheckte, von denen vielleicht *fünf* braun-weiß und die *restlichen* schwarz-weiß gefärbt waren.

Sie besaßen *fünf* schwarze Katzen, *zwei* weiße, und *dreizehn* gescheckte, von denen vielleicht *fünf* braun-weiß und die *restlichen* schwarz-weiß gefärbt waren.

#### Katzen

schwarz	5
weiß	2
braun-weiß	5
schwarz-weiß	8

Abb. 16: Links: Willkürlich umgebrochener Fließtext im Vergleich zur Tabelle. Je nachdem wie der Zeilenumbruch erfolgt, verteilen sich die Quantitäten im Text anders. Mitte: Im Fließtext wird die Quantität ›fünf‹ durch den Zeilenumbruch willkürlich vom Kontext ›braun-weiß‹ getrennt. Rechts: In der Tabelle ist ein sofortiger Überblick über die Quantitäten möglich, jedoch geht die Unschärfe des Wortes ›vielleicht‹ verloren (Autor).

In der Tabelle hingegen wird der Informationsfluss aufgespalten (Abb. 16). Die entstehende Aufspaltung, die Spalte, lässt sich am ehesten über die Kolumne, die *columna* (lat. für Säule, Pfeiler) etymologisch aufschlüsseln, jene vertikale, lotrechte architektonische Formation, die ein Dach oder Gewölbe trägt. Sie grenzt die Informationsobjekte voneinander ab und versiegelt sie in Zellen beziehungsweise legt sie in offen in Feldern.<sup>103</sup> In der Zeile, der »abgeteilten Reihe« (Duden), aus dem 9. Jh. von der mittelhochdeutschen *zīla*, mit der Bedeutung Reihe, Linie, Gasse abstammend, sammeln sich horizontal organisiert die Informationsobjekte zur Entität.

Im Unterschied zum Fließtext entsteht die Anschaulichkeit nicht aufgrund semantischer Eigenschaften wie Sinnlichkeit oder Lebendigkeit – diese werden in der Tabelle weitestgehend reduziert. Sie ist vielmehr Ergebnis der Ausbreitung und Verteilung der Inhalte im zweidimensionalen Tabellenraum. In

<sup>102</sup> Ausführlich siehe *The Domestication of the Savage Mind* (Goody 1977, 74–111), insbesondere S. 84, 88. Vgl. auch *Datenbanken* (Weingarten 1994, 159); Abweichend dazu argumentiert Eleanor Robson, dass sich in den frühesten bekannten Aufzeichnungen der Sumerer keine Tabellen fänden. »Die 5600 Dokumente aus dem Uruk des späten vierten Jahrtausends v.u.Z. und seiner Nachbarsiedlungen ›diente dem Zählen einer umfangreichen Administration, die Büros für Fischerei, Herdentiere und Tierprodukte, Verwaltung der Felder, Getreideproduktion und –verarbeitung und für Arbeitsangelegenheiten umfasste‹. Sie waren nicht als Textzeilen geordnet, sondern in Felder oder ›Kasten‹, die jeweils eine Sinneinheit oder Buchungseinheit repräsentierten« (Robson 2003, 21). Robson legt dar, dass die Mehrzahl der Dokumente die Aufzählungen nicht in Tabellenform abhandelt, und sie belegt als erste Tabelle eine Datensammlung aus Puzrtsh-Dagan zum Tierbestand von 2028 vor unserer Zeitrechnung (ebd. 24).

<sup>103</sup> Während ›Feld‹ den räumlichen und offenen, landschaftlichen Aspekt betont, verweist ›Zelle‹ auf den abgrenzenden, geschlossenen Charakter der Gefängnis- oder Klosterzelle. Beide Begriffe werden in der vorhandenen Literatur unscharf verwendet.



diesem Sinne erlaubt die Tabelle eine maximale Ökonomie der Darstellung, indem die Semantik aufgespalten wird und eine Reduktion auf ›reine‹ Daten erfolgt.<sup>104</sup>

Dieses wahrnehmungsökonomische Motiv setzt sich fort: Tabellen werden erstellt mit dem Ziel, Evidenz zu erzeugen. Dazu bedarf es neben der Tabelle selbst, der Informationsobjekte mit denen sie gefüllt wird. Dabei ergibt sich eine doppelte Bewegung: es sind einerseits die Informationen aus deren Perspektive es einen passenden Ort in der Tabelle zu finden gilt und andererseits ist es die Tabelle, die definiert, was überhaupt eingetragen werden kann. Die Tabelle mit ihren Einschlüssen und Ausschlüssen von Realität, der Präsenz abwesender Dinge, ist somit nicht nur Informationsspeicher sondern ebenfalls ein expliziertes Informationsmodell.

Die *constructio* stellt Struktur und Rigidität zur Verfügung. Das Tabellenraster dient dabei der Homogenisierung des gegebenen Inhomogenen. So wird das Datum zu Daten. Wesentliches Merkmal dieser *constructio* ist ihre Erweiterbarkeit, jederzeit können neue Spalten und Zeilen eingefügt werden. Diese Modularität erwächst aufgrund der Rechteck-Form der Zellen maximal effektiv, denn zwischen den einzelnen Zellen entsteht kein zusätzlich zu füllender Raum. Untereinander liegende Zellen sind durch die gleiche Breite gekennzeichnet und nebeneinanderliegende Zellen durch die gleiche Höhe. Gleichzeitig ist die *constructio* der Tabelle durch größtmögliche Abstraktion gekennzeichnet. Hier strukturiert die graphische Lösung ein abstrahierendes, kategoriales Denken mit reduzierten Informationsobjekten vor.

Der konstruktive Darstellungsrahmen erzeugt eine Funktionalität, die sich in den algorithmischen und (Wissens-)Operationen, die auf die Inhalte ausführbar sind, niederschlägt. Die Tabelle wird zu einem Denkwerkzeug, welches für jeden Sachverhalt neu modelliert und konstruiert wird. Sie dient dem Lösen von Problemen und dem Steuern von Prozessen.<sup>105</sup>

#### FORMATION – IN FORMATION – INFORMATION

Ab Mitte des 16. Jahrhunderts, dem Einsetzen der Frühmoderne, ist eine zunehmende Formalisierung der Informationsverwaltung in Mitteleuropa zu beobachten: Ein erster Schritt ist gekennzeichnet durch den gebundenen Bibliothekskatalog, in welchem in Listenform, das heißt Zeile für Zeile jeweils ein Buch mit Autorennamen, Titel, Verlag, Jahr, Standort, Exzerpt etc., in immer gleicher Reihenfolge vermerkt war. Die typografische Differenzierung der einzelnen Positionen, kennzeichnet einen weiteren Akt der Formalisierung, zum Beispiel wurde der Autorennamen fortan durchgängig in Großbuchstaben angegeben. Später werden die Positionen voneinander abgetrennt, indem man die Informationen auf einzelnen Zetteln und in der Folge auf Karteikarten aufteilte. Diese zunehmende Formalisierung setzte sich in der graphischen Abtrennung der Felder fort, die als Vordruck auf Karteikarten den genauen Ort und die maximale Größe des Eintrags vorgaben (vgl. Krajewski 2011).

Dies ist bereits das Formular, dessen leere Felder dazu auffordern, gefüllt zu werden. Eine Aufforderung zur Vollständigkeit, die in der Tabelle ähnlich implementiert ist. Die leere Zelle impliziert eine defizitäre Logik, das Vakuum eines zu füllenden Wissens- oder Datenraums. Diese defizitäre Logik, ein »Struktureffekt« (Steiner/Benjamin 2008, 283), ist Anlass für prozessorientiertes Tabellieren. Es gilt, die

---

<sup>104</sup> »Die Speicherung von Daten in bestimmten Strukturen, wie z.B. in Tabellen, ist eine andere Form der Übersetzung semantischer Informationen in Syntax« (Burkhardt 2015, 174) merkt Burkhardt zum Verhältnis von Daten und Struktur an; Die Vorstellung der ›Reinheit‹ von Daten gilt es anderswo zu diskutieren, siehe auch *Raw Data is an Oxymoron* (Gitelman 2013); Beschreibend (Grier 2005, 136).

<sup>105</sup> Zum Werkzeugcharakter: Krämer spricht in Bezug auf das Diagramm von einem »konstruktiven Werkzeug zum Problemlösen und Beweisen« (Krämer 2010a, 182) und Goody ordnet die Tabelle ein als »ein einfaches graphisches Werkzeug [...] für die Forschung über [...]« (Goody 1977, 67f.).

erst durch das Tabellengitter produzierten Leerräume zu beseitigen, also eine Prozess in Gang zu setzen, der all jene Handlungen beinhaltet, die zum Füllen der Zelle und damit zur Beseitigung des visuellen Malus der Leere beitragen.<sup>106</sup> Dies macht die Tabelle zu einem Werkzeug, wie es der Fließtext nicht leisten kann. Es ergibt sich ein zwingendes Wechselspiel aus gefüllten und leeren Zellen – für die Wissensgenerierung bedarf es der einen wie der anderen. Auf die Leere wird zurück zu kommen sein. Vorher jedoch wollen wir die Frage der Constructio weiter erforschen. Auf welche Weise wird diese Dialektik aus ›Leer⟨→Gefüllt⟨ in der, durch die vertikalen und horizontalen Linien erzeugten, grafischen Struktur der Tabelle wirksam? Wie entsteht im Lesen Referenz? Diese Fragen sollen uns zuerst beschäftigen, denn sie geben Aufschluss über mögliche Operationen, zu denen unter anderem auch die Leere zählt.

Einander naheliegende Inhalte ergeben Zusammenhänge, strukturiert durch die Leserichtungen der Spalte, oben-nach-unten und der Zeile, links-nach-rechts. Vergleicht man die Tabelle mit dem Diagramm, fällt auf, dass in jenem die räumlichen Beziehungen wesentlich offener sind. Die frei verteilbaren Datenpunkte werden im Verhältnis zu den rechteckig fix angeordneten Koordinatenachsen gelesen. Die Formen ergeben Flächen, Linien, Kurven, eine Vielfalt, die das Diagramm als Werkzeug zur Wissenskonstitution näher zu Kartographie situiert. Die graphische Zuspitzung der Tabelle auf zwei hauptsächlich Leserichtungen hingegen erzeugt zwei strukturelle Relationen.<sup>107</sup>

Erstens, ›Zeile⟨: Jeder Datensatz wird vom folgenden abgegrenzt, indem eine weitere Zeile eröffnet wird. Die Zeile stellt die Entität zueinander gehöriger Informationsobjekte dar. Ihr erster Eintrag weist der Entität einen Namen oder eine Adressierung (z.B. eine Signatur) zu. Die Zeile spannt eine ›strukturelle Relation⟨ der einzelnen Informationsobjekte auf. Pro Zeile beziehen sich die verschiedenen Attribute auf ein spezifisches Objekt. Die Tabelle eines Geologen ermöglicht beispielsweise pro Zeile die Darstellung spezifischer Attribute (Gesteinsart, Gewicht, Farbe) für ein Objekt (einen Stein). Für viele Objekte (viele Steine) werden die Zeilen untereinander geordnet.<sup>108</sup> Die Attribute in der strukturellen Relation der Zeile sind durch eine Boolesche Logik des UND miteinander verknüpft.

Zweitens, ›Spalte⟨: Die Spalte kann eine ›strukturelle Relation⟨ über ein spezifisches Attribut erzeugen. Im Spaltenkopf ist ein kategorialer Name verzeichnet. Von oben-nach-unten trifft der Leser auf immer das gleichbleibende Attribut (z.B. Gewicht) verschiedener Entitäten (z.B. verschiedener Steine), beziehungsweise auf die Leerstelle noch zu ergänzender Daten. Die Spalte erzeugt Konsistenz und der durchgängige Datentyp in einer Spalte ermöglicht, so es sich um Ziffern handelt, algebraische Operationen wie zum Beispiel Summe oder Mittelwert. Kategoriale Attribute stellen durch die Spalte hindurch topologische Operationen wie Sortierung, Filterung (im Sinne von Ausschluss) und Gruppierung zur Verfügung (Abb. 17).

---

<sup>106</sup> Fragil, obgleich dem Anschein nach stabil, bleibt umgekehrt, was die ausgefüllten Zellen im Vergleich zu den leeren behaupten: Vollständigkeit. »Gerade durch das Vorhandensein von Lücken und Leerstellen in einer Tabellengeschichte wirkt umgekehrt ein lückenloser Abschnitt so, als ob man über das notwendige Wissen vollständig verfüge« (Steiner/Benjamin 2008, 288).

<sup>107</sup> Der Begriff Relation wird hier im Sinne Charles S. Peirce als Beziehungen zwischen Informationsobjekten gebraucht (Krämer 2009/2010 Vorlesung 12), und weicht damit von der mengentheoretischen Verwendung im Zusammenhang mit relationalen Datenbanken in (Codd/Edgar F. 1970) ab.

<sup>108</sup> Es gibt eine Wendung: Die zweispaltige Tabelle, in der pro Zeile ein Gegensatzpaar dargestellt wird zum Beispiel ›süß | sauer⟨, ›kalt | heiß⟨. Hier spielen bezieht sich die Relation auf etwas Trennendes, Abgegrenztes.

	Attribut	Attribut	Attribut	
Entität				
Entität				
Entität				
Entität				Tupel

Gestein kategorial	Gewicht in kg Datentyp	Farbe kategorial
Magma	12	grau
Quarz	6	orange
Marmor	6	grau
Sandstein	24	ocker

Sortierung	Summierung Mittelwert Sortierung	Sortierung Filter Gruppierung
------------	--	-------------------------------------

Abb. 17: Da im kategorialen Attribut Gestein nur verschiedene Werte vorhanden sind, ist allein Sortierung als Operation möglich (entlang einer ODER-Logik). Im kategorialen Attribut Farbe hingegen tritt der Wert grau mehrfach auf und neben Sortierung werden auch Gruppierung (UND/ODER Logiken) und Filter (NOT Logik) verfügbar (Autor).

Beide Richtungen (links–rechts; oben–unten) ermöglichen, neben algorithmischen Operationen auf ihren Inhalt, Wissensoperationen des Betrachters. Zeile und Spalte unterliegen jeweils einer Logik der Nähe/Entfernung und des Oben/Unten, die es ermöglicht, Differenzen und Ähnlichkeiten über das gesamte Tableau verteilt zu erkennen und somit neues Wissen zu generieren. Die Tabelle erzeugt eine grammatikalische Ordnung, deren Regeln bestimmen, welche Operationen auf die Relation ausgeführt werden können. Diese grammatikalische Ordnung ist gekennzeichnet von einer »Kausalitätsanmutung« (Schmidt-Burkhardt 2012, 126), welche Evidenz im Sinne unmittelbarer Anschauung, erzeugt. Drittens, »Dynamische Relationen«: Durch den Blick des Betrachters ermöglichen Operationen auf Zellen oder auf Mengen von Zellen. Diese Relationen werden durch das Verknüpfen verschiedener Zellen durch die Struktur hindurch gebildet. Sie lassen sich in kategorial bestimmte Wissensoperationen und algorithmische Operationen aufteilen.

Die »kategorialen Operationen« auf dynamische Relationen zeichnen sich durch Richtungswechsel des Blicks, Wechsel zwischen Detail und Über-Blick und Zugriff auf Informationsobjekte jenseits von Zeile und Spalte, sowie den überprüfenden Blick in die Zeilen- und Spaltenbezeichnung bzw. die Legende, aus. Damit verändert sich auch das Verhältnis, in dem Informationsobjekte in Bezug auf Entität und Attribut betrachtet werden: In der Zeile liegt eine starke Betonung auf der Entität, in der Spalte liegt eine starke Betonung auf dem Attribut. Bei Wissensoperationen dynamischer Relationen liegt die Betonung weder auf dem einen noch dem anderen. Dies lässt vermuten, dass es entweder keine Betonung gibt, oder der Schwerpunkt des Erkenntnisprozesses auf den Informationsobjekten (dem Zelleninhalt) liegt. Die kategorialen Operationen umfassen Sortierung, Gruppierung, Filterung und können mit den logischen Kalkülen der booleschen Algebra (z.B. UND, ODER, NICHT) verarbeitet werden.

Für »algorithmische Operationen« auf dynamische Relationen gilt zusätzlich: Die einzelnen, in den Zellen voneinander abgegrenzten Werte, werden durch einen Algorithmus verknüpft, zum Beispiel summiert, multipliziert oder den logischen Kalkülen der booleschen Algebra unterworfen. Dieser Prozess generiert Ergebnisse, die wiederum in der Tabelle in spezifischen Zellen niedergelegt werden. Nutzern der Tabellenkalkulation Excel zum Beispiel werden diese als SUMME (A1+C4) geläufig sein, wobei hier eine Adressierung greift und zwar der Zellen A1 (Spalte A, Zeile 1) und C4 (Spalte C, Zeile 4), deren enthaltenen Daten summiert werden.<sup>109</sup>

<sup>109</sup> In Tabellenkalkulationssoftware wird die Adressierung der Zellen durch die Software zur Verfügung gestellt.

Wir verfügen nun über eine grundlegende Idee, welche Lese-Operationen, die gleichzeitig wissensbildend sind, zur Verfügung stehen. Die Frage der Operationalität lässt sich jedoch kaum diskutieren, ohne sich der Adressierung zu widmen, die mit Kittler gesprochen, einer der kulturtechnischen Vollzüge von Aufschreibesystemen ist und mit Althusser das Subjekt der Ansprache überhaupt erst konstituiert.<sup>110</sup>

#### ADRESSIERUNG UND TOPOLOGISCHE ORDNUNG

Strukturelle und dynamische Relation stellen die Information für Wissensoperationen zur Verfügung, indem sie diese in eine An-Ordnung bringen.<sup>111</sup> Wie wirkt sich die An-Ordnung der Informationsobjekte auf die Produktion von Evidenz aus? Wir können vermuten, dass das Tabellieren einer ebenso suchenden Choreographie des Blickes erfordert, wie jene Choreographie der Bibliothekare und Nutzer, die in den Regalreihen nach einem bestimmten Buch suchen. Ähnlich wie die nach Themen sortierten Regale einen Zusammenhang, eine Kategorisierung der dort enthaltenen Bücher erzeugen, so spielen Zeilen- und Spaltenbeschriftung sowie Sortierung und Gruppierung eine wesentliche Rolle für den ›Datenzugriff‹ des Auges. Nähe und Entfernung beschreiben sowohl in Bibliothek, zumindest jener, in der die Bücher systematisch aufgestellt sind, als auch Tabelle Zugehörigkeit oder Differenz. In Bibliotheken wandelte sich die Logik der Suche durch Annäherung (dem Wandern entlang der Buchreihen) mit der Einführung des Karteikatalogs zu einem spezifischen Zugriff anhand der Signatur. Diese eröffnete die Möglichkeit, verschiedene Katalogordnungen, zum Beispiel thematisch oder alphabetisch, auf die gleiche Signatur verweisen zu lassen, die wiederum zum Ort der Daten führt. Dabei verhält sich die Tabelle zur Datenbank wie das Regal zum Katalog. Die Datenbank schafft für die Tabelle eine neue Ordnung, durch Indizierung und Sortieroperationen, so wie der Karteikatalog verschiedene Zugriffe (z.B. alphabetisch, chronologisch, thematisch) auf die Regale erlaubt.<sup>112</sup>

In der Tabelle treten Werte und Adressen (Zeile, Spalte) getrennt voneinander auf. Die Werte sind als Informationsobjekte in der Zelle gespeichert und können unmittelbar ausgelesen werden. Diese Unabhängigkeit von Adresse und Speicherinhalt prädestiniert die Tabelle als Registratur- und Speichermedium und Operationsgebiet der Datenverarbeitung, da Werte verändert werden können, ohne den konstruktiven Rahmen der Tabelle zu stören. Semiotisch gesprochen, sind die eingetragenen Werte jeweils Index einer Realität, auf die sie verweisen. Sie referenzieren den abwesenden Zusammenhang, der jenseits der Tabelle liegt.

Die adressierenden Lesezugriffe beim Tabellieren treffen auf einen stabilen, konstruktiven Rahmen. Wenn aber der Rahmen stabil bleibt, muss es weitere Faktoren geben, welche das Ergebnis der ›Lektüre‹ beeinflussen können und überhaupt neue Referenzzusammenhänge ermöglichen. Dies sind topologische Operationen wie beispielsweise ›Sortieren‹ und ›Gruppieren‹.

---

<sup>110</sup> Zur Bedeutung der Adressierung äußert sich Friedrich A. Kittler in der vierten, vollständig überarbeiteten Auflage von *Aufschreibesysteme 1800–1900*, welche 2003, 17 Jahre nach der ursprünglichen Edition erschien, im überarbeiteten Nachwort: „Das Wort Aufschreibesystem [...] kann auch das Netzwerk von Techniken und Institutionen bezeichnen, die einer gegebenen Kultur die *Adressierung*, Speicherung und Verarbeitung relevanter Daten erlauben.“ (Kittler 1982, 501, Hervorh. F.H.); Vgl. das berühmte Beispiel Althusser zur Anrufung als Subjektkonstitution: »Hey, Du!« in (Althusser 1977, 142).

<sup>111</sup> Gabriele Schabacher konstatiert den grundlegenden Zusammenhang: »Adressen sind außerhalb einer Ordnung gar nicht denkbar« (Schabacher 2001, 20).

<sup>112</sup> Siehe auch Markus Krajewskis eindruckliche Beschreibung der Adressierungsprobleme in den Bibliotheken des 18. Jahrhunderts (Krajewski 2011, 32).

›Unsortiert/Ungruppiert‹ herrscht eine Logik der Eingangsreihenfolge, in der die Zeilen angelegt wurden: Die zuerst angelegten Datensätze befinden sich oben, die folgenden weiter unten. Sortieren/Gruppieren sind somit algorithmische Operationen, die eine neue symbolische Ordnung, in Abkehr von der willkürlichen Reihenfolge des ursprünglichen Dateneingangs herstellen. »Durch die Psycholinguistik wissen wir, dass ›oben‹ und ›unten‹, ›links‹ und ›rechts‹ implizite Werte darstellen. So wird die vertikale Achse zumeist positiv bewertet. Je höher sich etwas darauf befindet, desto besser schneidet es ab. Das gleiche gilt umgekehrt auch: Je tiefer etwas liegt, desto negativer wird es eingestuft (vgl. Tversky 1995:47)« (Schmidt-Burkhardt 2012, 32). Somit ergeben sich durch Sortierung und Gruppierung neue Lesweisen einer Informationsmenge.<sup>113</sup>

Die ›sortierten Zellen‹ nehmen ihre jeweilige Zeile, ihre Tupel, an den neuen Ort, der ihnen durch die Sortierung anhand Attributs zugewiesen wurde. Die gesamte räumliche Struktur der Daten wird anhand des Sortierkriteriums neu organisiert. Die Adresse der jeweiligen Zeile verschiebt sich, und aus der 9. Zeile kann die 25. werden. Dabei wirkt die jeweilige Zeile als Ordnungszusammenhang, der systematisierend wirkt – als Relation. Die Relation der Zeile, die im Prozess des Sortierens immer bestehen bleibt, erlaubt das Vor-Auge-Stellen eines Zusammenhanges, eine Wertung der Relation im Verhältnis zu den anderen Relationen.

Das ›Gruppieren‹ dient vor allem dem Klassifizieren in verschiedene Kategorien, wobei das Verbindende eine gemeinsame Eigenschaft, Funktion oder Referenz sein kann. Das Gruppieren bezieht sich auf die verschiedenen Datensätze (Zeilen), die anhand des Gruppierungsmerkmals innerhalb der Tabelle neu geordnet werden und in Differenz zu den anderen Gruppen stehen. Auch hierbei ändert sich die ursprüngliche Adressierung, das Auge muss andere Zeilen abtasten als vorher, einhergehend mit dem Versprechen auf Erkenntnis durch die räumliche Verdichtung der Gruppe.

Das Tabellenraster lässt sich als Konstruktionsmittel zur Darstellung und Herstellung eines diskursiven Feldes beschreiben. Es dient dazu, die Gegenstände des Diskurses in ein Spezifikationsraster zu stellen, sie zu adressieren – eine Figuration, welche die Gegenstände des Diskurses zueinander positioniert, sie gegeneinander abtrennt, miteinander gruppiert und schließlich Ableitungen ermöglicht. Dieses diskursive Feld ist gerade durch die *constructio* der Tabelle nicht-identisch mit den ursprünglichen diskursiven Feldern, aus denen die einzelnen Gegenstände entnommen und in die Tabelle gestellt wurden. Das Spezifikationsraster ist einer der Modi, in dem die Gegenstände des Diskurses in Formation gebracht werden.

Es wird deutlich, dass das Tabellieren nicht allein einer Logik des Aufaddierens von Informationen folgt. Vielmehr entfalten sich mannigfaltige Beziehungen zwischen den in den Feldern aufgeführten Informationsobjekten, die durch Blick des Betrachters erst konstituiert werden. Die Tabelle wird zu einem der Werkzeug des Denkens, das Tabellieren zum Hantieren mit noetischen Gegenständen. Sie konstituiert sich als Speicher/Registrar mit Lese- und Schreibzugriffen, als räumliche Konfiguration und *Constructio* und als Spezifikationsraster diskursiver Formationen.

---

<sup>113</sup> Die Einführung der Tabellensoftware Excel hat dabei zu einer Verschiebung geführt: War die in der Tabelle etablierte Ordnung in der Ära des Buchdrucks allein vom Autoren der Tabelle zu etablieren und auf Papier dauerhaft fixiert, so erfuhr diese fixierte Autorität eine Verschiebung zugunsten der interaktiven Nutzung von Tabellen in der entsprechenden Software. Es bleibt jedoch zu betonen, dass in Excel eine semi-fixierte Form der Tabellennutzung vorrangig ist.

### 2.2.1.3 Diagrammatiken – Zeitordnung und Raumordnung

»Die Fläche ist ein Schatten – ein Ort ohne Tiefe, eine Nacht ohne Sterne« (Meynen 2004, 193).

Die Tabelle ist, wie Diagramme und Karten, von verschiedenen Autoren im Feld der Diagrammatik verortet wurden. (Schmidt-Burkhardt 2012, 11–15; Campe 2002, 239–276). Die kurze Antwort auf die Frage, warum die Tabelle überhaupt als diagrammatisch untersucht werden sollte, liegt in der räumlichen Organisation der Tabelle.<sup>114</sup> Ihre Elemente können vom Auge simultan erfasst werden. Gleichzeitig erzeugt die Verwendung graphischer Elemente (Linien und Zwischenräume) und Ordnungen (oben/unten, links/rechts) einen bildlichen Charakter. Bei der Tabelle, dem *Tableau*, handelt es sich um eine Abbildung.

Zwischenräumlichkeit, so Dieter Mersch, sei ein wesentliches Element diagrammatischer Strukturen. Diese erlaube es, durch räumliche Beziehungen einen »piktoralen Sinn« (Mersch) zu erschließen, der über die Diskretheit der Schrift hinaus weise. »Das bedeutet, Spatialität überhaupt als leitendes Prinzip des Diagrammatischen auszuweisen« (Mersch 2006, 105). In Bezug auf Diagramme schreibt er den räumlichen Anordnungen eine epistemische Rolle zu. Wie aber entstehen diese wissensbildenden Qualitäten? Krämer sieht die Übertragung von »Zeitfolgen in Raumstellen« (Krämer 2010a) als epistemisch folgenreichste diagrammatische Form der Übersetzung. Als Beispiel dafür sei die Aufmerksamkeit auf Colberts Inspektoren<sup>115</sup> gelenkt, die im linear organisierten Text ihrer Reiseberichte zwar das Verfließen der Zeit festhielten. Aber erst durch das Tabellieren gelang es, Ort und Zeit als Daten zu konstituieren, indem sie aus dem Fließtext erhoben wurden und einer Datenverarbeitung, z.B. dem Sortieren, Gruppieren oder Summieren, zugeführt wurden.<sup>116</sup>

Es wäre eine Fehlannahme, die räumliche Organisation als überhistorisch auszuweisen. Im Gegenteil – über die Jahrhunderte können wir beobachten, dass die *constructio* diskursiven Praxen unterlag: So wandelte sich beispielsweise die übliche Anordnung der Jahreszahlen von der Horizontalen zur Vertikalen und die Aufteilung der Spalten wurde zunehmend rigider gehandhabt. Wurden anfänglich bei Bedarf Spalten inmitten der Tabelle hinzugefügt oder beendet, zum Beispiel in Phrygio 1534 (Abb. 18), so wurde später die Spaltenordnung zunehmend strenger, durchgehend ab der Kopfspalte etabliert (Steiner/Benjamin 2008, 87–97). Neben dem Wandel solch grundsätzlicher Konventionen wie der Leserichtung, changierte auch der Umgang mit der Verräumlichung von Zeit. Ausgangspunkt war das Anliegen der frühneuzeitlichen Historiker, Erzählungen welche auf verschiedenen Zeitrechnungen basierten, zusammen zu führen. So galt es, den römischen, profanen Kalender (*Ab Urbe Condita, dt: Seit*

---

<sup>114</sup> Peirce bezeichnet alles schlussfolgernde Denken als diagrammatisch (Peirce 1976, 314) und (Shin 2002, 20). Bredekamp et al. kennzeichneten das Feld der Diagrammatik als Text-Ikon-Hybride, und Sybille Krämer markiert die Kombinationen von Graphemen und Text im Diagramm als »graphische Zwischenwelt«, in der Allgemeines sinnlich anschaulich gemacht und Begriffliches verkörpert wird. Diese spezifische Form der Darstellung bilde eine »Gelenkstelle zwischen Denken und Anschauung« (Krämer 2010b); Birgit Schneider fasst unter Diagramme jene visuellen Konstrukte, die es erlauben »Relationen und Muster im erforschten Gegenstand durch eine Methode der Verräumlichung zu erkunden« (Schneider 2014, 66).

<sup>115</sup> Finanzminister Jean Baptiste Colbert hatte unter König Ludwig XIV. ein neues, proto-gouvernementales System der Regierung etabliert, bei dem der König durch statistische Methoden über den wirtschaftlichen Zustand seines Reiches informiert war und daraus entsprechende Schlüsse der Steuerung ziehen konnte. Die Inspektoren wurden ins Land ausgesendet, um Daten zu erheben. Ausführlich dazu siehe (Soll 2009).

<sup>116</sup> Im Zuge dieser Erhebung erfolgt zudem eine Aufteilung quantitativer und qualitativer Daten, indem diese in verschiedene Spalten verteilt werden.

der Stadtgründung [Roms]) zu synchronisieren mit dem christlichen, an biblischen Ereignissen orientierten Kalender *Annus Mundi*.

The image shows two pages from a historical chronicle, likely a calendar or a list of events. The left page is titled "CHRONICON" and the right page is titled "PAVLI CONSTANT. PHRYGIONIS.". Both pages feature a vertical timeline of events, with the year 186 at the top left and 285 at the top right. The timeline is organized into columns, with the central column containing the years. The text is in Latin and includes names of emperors and events, such as "CASSIANVS Hierofola", "VALENTINVS praeser Augustus", "CHRISTVS in Iudaea corpora natu", "CERDON duos dies postea, alio", "TITVS MARIANVS", "CARPOCRATES", "ANTONINVS", "T. ANTONINVS Pater Augusti. III. RVFINVS.", "ABELIVS AVRELIVS Caesar.", "VALENTINVS Hierofolitus", "P. PLAVTIVS", "IVL. SEVERVS.", "IVSTINVS philofophus Apologien", "compositus", "TORQVATVS II. HERODES AELIUS.", "T. ANTONINVS Pater Augusti. III. Pater Augusti. III. AE LIVS ANTONINVS Caesar II.", "LARGVS. MESSALINVS.", "ORPHITVS. PRISCVS.", "MAXIMVS. GORDIANVS.", "CLAVS II. SEVERVS.", "TORQVATVS III. IVLIANVS.", "GALLICANVS. VERVATVS.", "MARCVS ALEXANDER Aegypti Epifcopus.", "EVMERIDIS Iosephus, Epifcopus.", "AFOLLONIVS Bithyni philofophus.", "CRESCENS Cyprius philofophus in fine persecucionis monachus qui CHRISTI STI munus suscepit non fuit.", "Fuit temporibus illis quidam Marcus Colobryade quorundam hereticus hunc Epifcopum et Titum et C. C.

Abb. 18: Vertikale Tabelle: Die Zeit verläuft von links nach rechts. Spezifisch auch die oberen vier Zeilen mit jeweils eigenen Zeitrechnungen, die zueinander synchronisiert wurden: *Annus Mundi* (Weltjahre), Diluvii (Seit der Sintflut), a nato Christo (Seit Christi Geburt), Olymp (griechische Zeitrechnung), (Phrygio 1543, 284–285).

Frühe Historiographien orientierten sich vielfach an Perioden, die durch die Lebensdaten kirchlicher oder weltlicher Herrscher vorgegeben waren, während erst im Zuge der Aufklärung eine Geschichtsschreibung anhand abstrahierender Zeitmaße, zum Beispiel der Jahrzehnte, erfolgte. Die tabellarische Form erwies sich als Anlass und Medium der Synchronisation zugleich.<sup>117</sup> Steiner argumentiert zu Recht, dass in der Tabelle ein euklidischer Raumbegriff vorherrsche, denn es gebe in graphischer Hinsicht, im Unterschied zum Diagramm, keine Beugung, Krümmung oder Verformung. Vielmehr seien die Zeit- und Raumpalten der Logik des Newtonschen Gitters unterworfen, eine Matrix, welche die Weltereignisse, Orte, Verläufe oder Wendepunkte in eine lineare und gerade Form bringe. Er gehe daher davon aus, dass Tabellen formal den absoluten Raum und die absolute Zeit gegenüber einer relativen Auffassung von Raum und Zeit, wie sie im 20. Jahrhundert durch Albert Einstein, Kurt Gödel und weitere entwickelt wurde, prädestinieren – eine »Euklidisierung der Geschichte« (ebd. S. 280). Hier möchte man einwenden, dass es beispielsweise durch unterschiedliche Zeilenhöhe oder die Verwendung unregelmäßiger Zeitskalen sehr wohl zu einer Verzerrung der Zeitdimension kommen kann.<sup>118</sup>

<sup>117</sup> Siehe Abbildung Phrygio, dort werden folgende Zeitrechnungen synchronisiert: *Annus Mundi* (Weltjahre), Diluvii (Seit der Sintflut), a nato Christo (Seit Christi Geburt), Olymp (griechische Zeitrechnung).

<sup>118</sup> Gemeint ist die vertikal orientierte Tabelle, in der die Zeit von oben nach unten verläuft – analog wäre das die Spaltenbreite in der horizontal organisierten Tabelle.

Und jenseits dieser diagrammatischen Dimension trifft das Tabellieren ganz praktisch auf das Problem, Zeit und Ereignisse einhundertprozentig zu synchronisieren. Das sieht auch Steiner: »In der Tabellengeschichte kommen relationale und absolute Vorstellungen von Raum und Zeit gleichzeitig zur Geltung, ohne dass sich die Autoren eine solche Weltsicht bewusst zurechtgelegt haben« (ebd.).

Es bleibt festzuhalten, dass die Raumordnungen der Tabelle diskursiven Veränderungen unterliegen und dass es wird daher angezeigt ist, diese genauer zu untersuchen. Dies geschieht im Abschnitt 2.2.2. *Tabellarische Praktiken*. Vorerst aber sollen nach dem Durchgang durch die diskursiven Verhältnisse von Aufzählung und Erzählung, Konstruktion, Formation und Information, Adressierung und Topologien, sowie Raum- und Zeitordnung, abschließend die damit ermöglichten Verfahren von Wissensoperationen beschrieben werden.

#### 2.2.1.4 Wissensoperationen – Den tabellaren Raum durchschreiten

»Das Buch ist voll von Tabellen. Ich hasse Tabellen, die jemand anderes als ich gemacht hat. Wenn ich Bücher aufschlage, da sind Tabellen drin, das finde ich zum Kotzen. Ich habe mich ganz lange dazu entschließen müssen, selbst welche anzufertigen, aber ich glaube wirklich, dass sie zur Klarheit beitragen und werde ihnen das an dieser Tabelle ganz besonders drastisch demonstrieren« (Winkler 2015, 50:48 min).

Sybille Krämer beschreibt die diagrammatische Form als »Gelenkstelle zwischen Denken und Anschauung« (Krämer 2010b, 176) und wirft damit die Frage auf, entlang welcher spezifischen Materialität die ›Gelenkstelle‹ Tabelle eine Operationalisierung von Wissen ermöglicht. Konkret: Wie lässt sich die Metamorphose von der realen Schreiboberfläche zum zweidimensionalen idealen Ebene in der Tabelle als Denkraum spezifizieren? Es lohnt dieser Frage nachzugehen, weil das Nachdenken darüber deutliche Hinweise darauf vergeben mag, wie das Tabellieren kulturtechnisch für (relationale) Datenbanken konstituierend wird. Möglich wird eine Beantwortung, indem die Anschauung als Vorgang begriffen wird, der sich durch eine Abfolge von gerichteten Vektoren des Blickes auszeichnet, welcher sich entlang des tabellarischen Rasters bewegt oder im Raster frei springt.<sup>119</sup> Denn es ist die Gitterstruktur, die den Tabellenraum in Richtungen organisiert. Sie lenkt den Blick, der die Syntax der Zeilen-Spaltenkonfiguration zusammen denkt mit dem semantischen Gehalt der Informationsobjekte.<sup>120</sup> Vier Operationen können beobachtet werden:

Erstens, ›Leere Zellen füllen‹: Das »Auge des Geistes« (Krämer 2010b, 180ff. in Anspielung auf Descartes) überprüft leere, zu füllende Zellen daraufhin, ob eine außerhalb der Tabelle gegebene Information darin überhaupt verortet werden kann. Wenn man die Attribuierung der Spalte und die Benennung der Zeile als semantisches Moment sieht, das durch die Tabellensyntax vom eigentlichen Informationsobjekt getrennt ist, aber in Beziehung zu ihm steht, so ist zu vermuten, dass das ›Auge des Geistes‹ jene semantischen Aspekte des Informationsobjektes in den Blick nimmt, die sich als Attribut oder Name

---

<sup>119</sup> Die folgende Konstruktion stützt sich auf Ideen, die allein phänomenologisch basiert sind, eine empirische Untersuchung, die diese Beobachtungen stützen würde, steht aus.

<sup>120</sup> Vgl. die Diskussion zu Syntax, Semantik und Pragmatik in (Burkhardt 2015, 191) und (Kosslyn 1989, 189).



einer Entität abspalten lassen. Das ›Auge des Geistes‹ sieht hier vom pragmatischen Gehalt der Zeichen ab, zugunsten der syntaktischen und semantischen Ver-Ortung in die Tabellenstruktur.<sup>121</sup>

Zweitens ›Gefüllte Zellen auslesen‹: In den mit Informationsobjekten belegten Zellen, sucht das ›Auge des Geistes‹ zur Beantwortung einer Fragestellung anhand der Kriterien, die durch Entität oder Attribut beschrieben sind, das geeignete Informationsobjekt, um es dem Denkprozess zur Verfügung zu stellen. Die vom semantischen Gesamtgehalt des Informationsobjekts abgespaltenen Aspekte ›Attribut‹ und ›Entität‹ stehen nun – verortet – dem Denkprozess zur Verfügung, um sie zu den verbliebenen semantischen Anteilen der Informationsobjekte ins Verhältnis zu setzen.<sup>122</sup> Aus der optosynchronen Zusammenschau der, durch die Syntax räumlich voneinander getrennten, semantische Anteile folgt die Pragmatik, der Wissensgehalt der Zeichen.

Drittens, ›Referenzialität‹: Aufbauend darauf werden topologische (Sortieren, Gruppieren), algebraische (Summe, etc.) oder logische (UND, ODER, NICHT) Operationen auf verschiedene Zellen ausgeführt, die neue Relationen schaffen. Sortierungen stellen eine hierarchische Relation der Informationsobjekte anhand von Kriterien her. Gruppierungen schaffen Relationen unter den Informationsobjekten, die zur jeweiligen Gruppe gehören und teilen die Informationsmenge in diskrete Untermengen. Algebraische und logische Operationen überlagern die Tabellensyntax mit einer spezifisch eigenen Syntax und schaffen damit neue pragmatische Zugänge.

Viertens, ›Gesamtheit überblicken‹: Zugunsten der Tabelle als Tableau weitet sich der Blick, verliert seine vektorielle Gerichtetheit, um die räumliche Verteilung mehrerer Informationsobjekte zu bestimmen.

Für diese Operationen sind verschiedene Methoden des vektoriellen Blicks beschreibbar, wobei die ersten eine Operation ›im Raster‹ evozieren: Das Verbleiben in der Zeile, also das horizontale Schreiten von Zelle zu Zelle und zwar ausgehend von der Beschriftung in Leserichtung; oder wenn man sich bereits tiefer im Tabellenraum befindet: entgegen der Leserichtung – das Durchschreiten der verschiedenen Attribute einer spezifischen Entität. Ebenso im Raster orientiert sich das Verbleiben in der Spalte, in einem einzelnen Attribut und das sukzessive Durchschreiten der Entitäten entlang der vertikalen Achse vom Tabellenkopf aus nach unten; oder das Rückschreiten oder -springen in höhere Zeilen der selben Spalte, auch entgegen der Leserichtung. Davon unterscheidet sich das freie Springen und kognitive Verknüpfen innerhalb der Tabelle, ›springend durch das Raster‹.

Während das Erfassen der Spaltenbeschriftungen und ggf. der Zeilenbeschriftungen im Raster bleiben, fallen andere Blicke aus dem Raster, sind aber eng mit dem Erfassen des Tabelleninhaltes verbunden. Dies betrifft den unmittelbaren (Bildunterschrift, Legende) und den weiteren Tabellenkontext: In welchem inhaltlichen Zusammenhang ist die Tabelle eingebunden, auf welche Textstelle im Fließtext bezieht sie sich.

Wir haben gesehen, wie das Tabellieren, als ›Gelenkstelle zwischen Denken und Anschauung‹ (Krämer), die noetischen Gegenstände einer sinnlichen Repräsentation zuführt, deren Spezifik die räumliche Syntax und damit die Verortung der semantischen Anteile ist. Als ›tabula‹, ›tavola‹ oder ›tableau‹ konstruiert die Tabelle einen Anschauungsraum, in dem dreidimensionale Qualitäten der Welt

---

<sup>121</sup> Während im Diagramm eine Kurve oder Linie mit geometrisch-zeichnerischen Mitteln durch Hand-Auge-Koordination verlängert, ein Leerraum also durch Konstruktion gefüllt werden kann, ist die Leere der Tabellenzelle eine andere: Entweder signalisiert sie die Abwesenheit von empirischen Daten, die noch zu erheben sind, oder sie signalisiert das Potenzial, die Daten algorithmisch aus bereits vorhandenen Daten zu generieren.

<sup>122</sup> Zum Beispiel wird aus »Die Farbe des Käfers ist rot« durch Aufspaltung des Semantischen anhand der Satzsyntax und Überführung in die Tabellensyntax die Entität ›Käfer‹, das Attribut ›Farbe‹ und das in der Zelle enthaltene Informationsobjekt mit dem verbleibenden semantischen Gehalt ›rot‹.

zweidimensional abgebildet werden. Es erfolgt eine abstrahierende Projektion des Realen. Diese Metamorphose von der realen Schreiboberfläche zur zweidimensionalen idealen Ebene als Denkraum erlaubt es, durch Neuverknüpfung der Informationsobjekte, die Potenziale eines spezifischen Informationsobjektes zu aktivieren. Das Potenzial steigt mit jeder weiteren Zelle, die vom ›Auge des Geistes‹ erfasst wird. Die Tabelle wird damit zum idealen Medium für die Speicherung und Verarbeitung von formatierten Informationsobjekten.

Nach dem Durchgang durch das dichte und umfangreiche Material soll an dieser Stelle zusammengefasst werden.<sup>123</sup> Die Tabelle zeichnet sich, wie wir gesehen haben, durch eine Reihe konstruktiv begründeter Qualitäten aus:

Erstens, ›Simultaneität‹: Das Erkennen von Mustern und Relationen geschieht durch den Blick, der die Oberfläche der Tabelle abtastet und die Informationsobjekte an ihre abgespalteten semantischen Elemente, die durch die Tabellensyntax in Zeile und Spalte (vor allem in den Spalten-Namen) verortet sind, rückbindet. Zur Simultaneität zählt auch, dass verschiedene Maßstäbe synchronisiert werden können, indem Spalten die gleichen Werte mit verschiedenen Einheiten (z.B. Lichtjahre, versus Kilometer) darstellen.

Zweitens, ›Räumlichkeit/Fläche/Richtung‹: Die zweidimensionale Schreiboberfläche wirkt auf ihre Informationsobjekte homogenisierend, sie hüllt die Informationsobjekte in Zellen ein und kontextualisiert sie mittels Lokalisierung und Adressierung. Sie formalisiert das Gegebene und bringt es in Formation. Das Raster evoziert im Tabellenraum Blick- bzw. ›Lese‹richtungen und führt in Sortierungen und Gruppierungen eine geometrisch angelegte Hierarchisierung auf.

Drittens, ›Hybridität‹: Das Bildhafte der Tabelle ergibt sich aus ihrer Überblicksfunktion im Sinne des Tableau. Dem Gesamtbild steht die in die Zelle gesperrte Sprache gegenüber. So kann eine Verschränkung von Ikonizität und Sprache konstatiert werden.

Viertens, ›Referenzialität‹: Referenzialität entsteht durch topologische, mathematische oder logische Operationen, mittels derer neues Wissen generiert wird. Das Tabellenraster stellt eine syntaktische Ordnung aus Entität und Attribut zur Verfügung und homogenisiert die Informationsobjekte zugunsten dieser Ordnung. Die Abspaltung von Entität und Attribut reduziert und abstrahiert die Abbildung des abwesenden Gegenstandes oder Vorganges, welcher dem Informationsobjekt in der Realität zu Grunde liegt und als Referenz dauerhaft verbunden bleibt. Entität und Attribut stellen strukturelle Relationen entlang von Blickvektoren zur Verfügung, die um dynamische Relationen des zwischen den Zellen springenden Blickes ergänzt werden.

Fünftens, ›Operationalität‹: Der diagrammatische, zweidimensionale Raum der Tabelle ermöglicht Operationen entlang der durch das Raster vorgegebenen Blickrichtungen oder springend durch das Raster. Kulturtechniken des Sortierens und Ordners treffen auf die Kulturtechniken des Lesens und Betrachtens.

Sechstens, ›Diskursivität‹: Tabellenpraxen wandeln sich abhängig vom gesellschaftlichen Kontext.

Siebtens, ›Medialität‹: Der tabulierte Raum kann als Speichermedium für vorhandenes Wissen dienen und es an andere Orte transportieren, als Übertragungsmedium, welches erlaubt, Realität in Informationsobjekte zu transformieren, sowie als operatives Werkzeug zur Erzeugung von Evidenz.

---

<sup>123</sup> Diese kategorialen Unterscheidungen haben sich vor dem Hintergrund der Diskussion in (Krämer 2010b, 185ff; Krämer 2010a, 35ff.), und der durch Latour für *Immutable Mobiles* entworfenen Kategorien (Latour 2006, 285f.) entwickelt.

Die vorgehende Diskussion kann in dem Sinne gelesen werden, dass man die Tabellarisierung einer Informationsmenge als Vorgang gesteigerter Abstraktion beschreibt. Bei aller Abstraktion, Komposition, Konstruktion haken jedoch immer auch Momente von Materialität und Handhabung ein, welche die Gegenstände den Sinnes-Registern zuführen. Diese gilt es es nun stärker herauszuarbeiten und zu diesem Zweck ist hier eine ganze Reihe von Fallbeispielen versammelt. Das Ansprechen der Sinnesregister bei zunehmender Abstraktion ist auch ein starkes Argument dafür, dass es im Fall der relationalen Algebra und der relationalen Menge des Tabellierens als eingeführte Kulturtechnik bedurfte, um die Register des Mathematischen auch Nicht-Mathematikern visuell zur Verfügung zu stellen.

### 2.2.2. Tabellarische Praktiken

Der folgende Abschnitt untersucht historische Praktiken des Tabellierens, um die bisherige Diskussion mit verschiedenen Praxen abzugleichen. Zeitlich setzt sie ein mit dem Erstarken der Wissensproduktion im 17. Jahrhundert und dem Übergang von der Manufaktur zu den industriellen Produktionsweisen. Ziel dieses Abschnittes ist es, eine große Bandbreite von Tabellenpraktiken aufzuzeigen, die disparat erscheinen mögen, in ihrer Verstreutheit aber auch ihre tiefgehende kulturtechnische Verankerung in der menschlichen Wissensproduktion anzeigen. Zeitliche Sprünge bleiben nicht aus, denn statt eine Chronologie vorzulegen, verfolgt diese Studie kategoriale Unterscheidungen: Wissensproduktion; Statistik, Organisation und Prozesse; und mathematische Tabellen. Diese Kategorien waren nicht von vornherein festgeschrieben, erst allmählich, im Zuge der Recherche und entlang der ausgewählten Praktiken schälten sie sich heraus. Durch sie hindurch zieht sich ein Set von Fragen: Welche Bedürfnisse der jeweiligen Zeit sollte die Tabellenproduktion erfüllen? An welche Grenzen stießen die Verfahren? Welche verallgemeinerbaren Operationen sind zu beschreiben und welche distinkten? Wie wirkt sich die Materialität der Tabelle und auf ihre Operationalisierung aus? Was verbindet die jeweiligen historischen Praktiken mit der Verwendung der Tabelle in der relationalen Datenbank?

#### 2.2.2.1 Wissensordnungen – Die Literatur des Mittelalters in nationalen Kategorien

Der Historiker und Hochschullehrer Gabriel Gottfried Bredow (1773–1814), der an den Hochschulen Helmstedt, Frankfurt/Oder und Breslau lehrte, versprach bereits in der Einleitung seiner *Weltgeschichte in Tabellen* die Übersicht über ein stetig wachsendes Wissensgebiet.<sup>124</sup> »Diese Tabellen erarbeitete ich zu dem Zweck, daß sie, ein Handbuch der Geschichte in tabellarischer Form, die Vortheile der synchronistischen Methode mit den Vorzügen der ethnographischen vereinigten, ohne daß die Erwartung, die man zunächst von Tabellen hat, getäuscht würde, eine leichte Übersicht der Geschichte einzelner Völker und der gesammten Weltgeschichte zu gewähren« (Bredow 1810).

Wichtige Daten werden, darauf wies Bredow hin, typografisch hervorgehoben, durch Einzüge, gesperrte und kursivierte Schrift sowie verschiedenen Schriftgrößen. Die Intention dieser typografischen Auszeichnung – und wir erinnern uns, dass Markus Krajewski diese als einen Schritt zunehmender

---

<sup>124</sup> Bredow und viele andere stehen mit ihren Tabellen in einer langen Tradition der »Wissens- und Gedächtnistabellen«, deren Genealogie Benjamin Steiner in seiner Dissertation *Die Ordnung der Geschichte – Historische Tabellenwerke in der frühen Neuzeit*. (2008) darstellt. Steiner diskutiert die Tabelle nicht allein als statisches Wissenswerkzeug, sondern belegt, wie sich die Ordnungsvorstellungen in der Frühneuzeit und mit ihnen die Tabellenform wandeln. So ist eine zunehmend empirische Hinterlegung der in Wissens- und Gedächtnistabellen aufgeführten Daten durch Quellenangaben zu beobachten, sowie eine Ablösung der Ordnungen von an konfessionellen Ereignissen orientierten, Päpste und Könige aufzählenden Historiographien zu opto-synchronen Chronologien, die breite Wissensfelder der Geistes- und Naturwissenschaften abbilden.

Formalisierung von Informationen kennzeichnet – schilderte Bredow dahingehend, dass der »Lehr-ling« zuerst die »wichtigsten Weltbegebenheiten« memorieren solle, dann die nach Völkern getrennten Informationen miteinander vergleichen und somit deren »Behalten« vertiefen möge und schließlich die Hervorhebungen als Darreichung für den Lehrer dienten, der nur einzelne Stichpunkte andeuten müsse, um das erlernte Wissen abzufragen. Zu diesem Ansatz passt, dass die Rückseiten der Tabellen nicht bedruckt waren und die Drucke nicht zwangsläufig als Buch gebunden, sondern auch auf einem Tisch ausgebreitet oder an die Wand gehängt, mithin als Tafel genutzt werden konnten.

Die zentrale Konzeption der Wissensorganisation in Bredows Tabelle *Litteratur des Mittelalters* (Abb. 19) zeigen Kategorien ethnografischer Zuordnungen. Somit wurden die schriftstellerischen Leistungen als ethnische/nationale Leistungen konstruiert und untereinander verglichen. Als solche wurden sie ins Verhältnis gesetzt zu den jeweils außen, links und rechts etablierten Kategorien der Politik und der Bildung. Augenfällig wird deutlich, wie sehr es sich um eine Verdichtung handelte, denn kaum ein Raum blieb ungenutzt, das ganze Blatt, jede Zelle wurde maximal gefüllt. Auffällig ist, dass es keine einheitliche Höhe der Zellen gibt – die Jahreszahlen sind pro Spalte einer trennenden horizontalen Linie folgend angegeben, so dass in benachbarten Spalten verschiedene Jahrhunderte neben einander liegen können, die Chronologie also nur grob der Zeile folgt.

Wicht. politische Begebenheiten.	Juden und Armenier.	Araber und Perfer.	Griechen, nach und nach Neu-Griechen.
400. die Westgothen, unter Alarich, in Italien; 416 in Spanien.	400. Juden in der Zerstreung, doch nicht ohne berühmte Schulen in Asien, u. voll Eifers für ihre Sprache.	400. Astrologie und Geschichte. Dichtkunst bildet die arab. Sprache, besonders den	400. Heidenische Gelehrte aus Alexandr. vertrieben. 400 <i>Oribasius</i> , Arzt in Alex. u. Sammler. — <i>Nemesius</i> <i>περι φουσκου</i> . — <i>Nonnus</i> Aeg., Dionysiaca.
450 Angeln u. Sachsen in Britann., Vandalen in Afrika, <i>Attila</i> , der Hunnenkönig.	400. <i>Misrob</i> , <i>Joh. Ekelensis</i> u. <i>Joseph Planensis</i> überl. d. Bibel ins Armenische.	500. Dialekt der <i>Korsischiden</i> . Poetische Wettstreite in Mecca, auf dem Platz <i>Oecadh</i> .	412 <i>Cyrillus</i> , Bisch. in Alex., ft. 444, geg. Ketzer. <i>Socrates</i> u. <i>Sozomenus</i> , hist. eccl.
476 <i>Romulus Augustulus</i> .		<i>Moallacat</i> oder <i>Modhabebari</i> . Sammlung der Preisgedichte. (7 sind noch vorhanden. Jones ed. London. 1783.)	430 <i>Codex Theodosianus</i> (ed. Ritter, Lips. 1789.) — <i>Synesius</i> Cyren., Orat. u. Epp. — <i>Zosimus</i> , hist. Caesl. — <i>Isidorus Pelusiota</i> , Exeget, Epp.
493 Theodorich, d. Orf.	462 <i>Moses</i> aus	600.	450 <i>Theodoret</i> Antioch., KV. — <i>Hierokles</i> Alex., <i>Aeneas Gaza</i> u. Schüler, Platon. Christ. <i>Syrianus</i> , Schüler des Neuplaton. <i>Plutarch</i> in Athen, Comment. in Aristot. <i>Metaph</i> , Lehrer des — <i>Proclus</i> Diadochus, Citpl. 412—485, Comment. in Platon. et Euclid. — <i>Marinus</i> , sein Schüler.
			477 verbrennt die kaiserl. Bibl. in Kistpl.

Abb. 19: Einzüge, gesperrte und kursivierte Schrift sowie verschiedenen Schriftgrößen dienen zur Strukturierung des Inhalts innerhalb der Zellen (Bredow 1810, 16–17, Ausschnitt, Linien verzerrt durch Scan).

Die Tabellen bildeten jeweils verschiedene zeitliche Epochen ab, beginnend mit Alte Litteratur bis 400 nach Chr und endend mit Litteratur der letzten zwei Jahrhunderte von 1600 bis 1800, in welcher, wie auch »LESSING« und »KANT«, ebenso in Versalien »GÖTTE, geb. 1749, Werther 1774, Iphigenie« vermerkt ist. Goethe wiederum, wir können uns vorstellen, wie er in seinem Zimmer vor dieser Tabelle steht und sie auswendig lernt, hatte sich Bredows Tabellenwerk in der dritten Auflage von 1810 zugelegt (Rößler 2014, 112f.).<sup>125</sup> Zwei Jahre später begannen Goethe und sein Freund, der Maler Johann Heinrich Meyer, die Tabellen Bredows mit kunsthistorischen Anmerkungen zu ergänzen. Als der Platz knapp wurde, klebte man einen weiteren Papierstreifen an die Seiten – ein untrügliches Zeichen für die Modularität von Tabellen. Goethe beauftragte sodann seinen Schreiber Johann August John mit der Verschriftlichung der Geschichte der Malerei und Skulptur. Diese Vorgehensweise demonstriert

<sup>125</sup> Zur Entstehung der Meyerschen Tabellen im zeithistorischen Kontext, siehe *Goethe und Johann Heinrich Meyers Tabelle zur antiken Kunstgeschichte* (Rößler 2014). Für eine Einordnung und Entwicklungsgeschichte der Wissenstabelle als didaktisches Mittel siehe (Steiner/Benjamin 2008, 164–235).



Literatur und Poesie«. <sup>128</sup> Jedoch nutzt er im Vergleich zu Bredow weniger typografische Merkmale zur Strukturierung des Textes innerhalb der Zellen. Lediglich die erste Zeile eines Absatzes enthält einen Einzug, wodurch innerhalb der Zelle eine deutlich listenorientierte Struktur erzeugt wird. Es gibt jedoch im Fließtext keine Kursivierung, Versalien oder ähnliches. Auch sind die Einträge pro Zelle teils sehr lang, besonders in jenen späteren Jahrhunderten (ohne Abb.). Dies schwächt die diagrammatischen Vorteile der tabellarischen Form zugunsten der Kontinuität des Fließtextes ab.

Da die Zeilenhöhen jeweils der Spalte mit dem meisten Inhalt angepasst sind, tritt zudem ein für Tabellen übliches Phänomen auf: bei großen Zeilenhöhen fallen leere Zellen umso signifikanter ins Auge und suggerieren verstärkt fehlende Informationsobjekte. Dabei bleibt unscharf, ob es sich um fehlende Überlieferung, oder die tatsächliche Absenz relevanter Ereignisse handelt. Die Zeitabschnitte in der mittleren Spalte | Zeitrechnung | sind nicht durch horizontale Linien voneinander getrennt und implizieren entgegen der sonstigen diskreten Feldstruktur eine Kontinuität der Zeit. <sup>129</sup> Ein ähnlich kontinuierlicher Raum entsteht in den schmalen Längsspalten. Durch die vertikale Schrift und das Weglassen trennender Linien gelingt es Meyer in der Tabelle, eine Kontinuität zu erzeugen und kunsthistorische Überschneidungen zwischen jenen Gebieten (Malerei, Plastik etc.) darzustellen, welche noch in seinem Buch streng chronologisch aber diskontinuierlich als Kapitel aufeinander folgten (Rößler 2014, 125f.). Der verbundene Zellenraum schuf eine Kontinuität, die auf die anliegenden Zellen und deren Interpretationsvarianten ›ausstrahlt‹. In dieser *constructio* verdeutlichen sich die diagrammatischen Qualitäten der Tabelle, da der spezifische graphische Aufbau wesentliche epistemische Qualitäten in sich birgt.

Zusammenfassend gesagt, dienten die von Bredow, Meyer und Goethe erstellten und verwendeten *Wissensordnungs*-Tabellen dem Memorieren von Informationen. Sie fanden ihren weiteren Einsatz als Denkwerkzeuge für das Verfassen neuer Texte, als schnelle Übersicht und Merkhilfe. Sie stehen in einer langen Tradition historiographischer Tabellen, die sich nach der Einführung des Letterndrucks zunehmend verbreiten und deren Auflage um die Jahre 1610 und 1700 ihre vorläufigen Höhepunkte erlebten (Steiner/Benjamin 2008, 131f.). Wissens- und Gedächtnistabellen wurden gemeinsam mit Kulturtechniken des Exzerpierens und Kategorisierens verwendet und sie dienten, als Ausgangspunkt weiterer Forschung und dem Generieren neuen Wissens. Wissenstabellen fordern eine Verdichtung und Drängung des Darzustellenden und etablieren den Leerraum als Ort des Ungewussten, welcher die Aufforderung das fehlende Wissen zu produzieren, evoziert. Das Ungewusste wird als eigene epistemische Qualität visualisiert.

#### 2.2.2.2 Statistik – Staatstabellen und Statistiken der Seidenproduktion

Markiert durch die Berechnung einer Sterbetafel in London von John Graunt 1662 und der *Politischen Arithmetick* (sic!) William Petrys, 1672 und genereller gesprochen mit dem Entstehen der Nationalstaaten ab Beginn des 17. Jahrhunderts, konstituierte sich ein neues Wissensfeld, in welchem Besitzverhältnisse, Bevölkerungszahlen und die staatlichen Strukturen erfasst und dargestellt wurden (vgl. Graunt 1662; Petty 1672). Es waren die ersten Regungen einer Gouvernementalität, jener Institutionen und Praktiken, die Menschen durch Regierungstechniken fügsam machen, sei es in Verwaltung oder im Schulsystem. Methoden der Datenerhebung, Statistik und Klassifizierung, die es erlaubten, soziale Verhältnisse als Daten zu erheben, wurden systematisiert. Die Aufzeichnungstechnik wurde zu einer

---

<sup>128</sup> Auffallend im Vergleich zu Bredow ist Meyers Abkehr von einer ethnologisch-nationalen Verortung der Kategorien.

<sup>129</sup> Horizontale Trennlinien würden die Jahrhunderte visuell viel stärker von einander als einzelne Einheiten abgrenzen.

Staatstechnologie, das Wissen über die Bevölkerung zum Staatswissen. Im Folgenden soll die Staatstabelle als spezifisches Medium des Regierens untersucht werden.

Die klassischen Aufgaben des Herrschers wandelten sich im 17. Jahrhundert. Bis dahin galt: dieser müsse weise und maßvoll sein, er müsse das Recht kennen und es anhand von Beispielen aufzeigen können und er müsse in der Lage sein, das Maß zu erkennen, anhand dessen Recht zu sprechen sei. Später wandelt sich diese Rolle: der Herrscher wird Regent. Während der Herrscher vor allem in der Lage sein musste, Recht zu sprechen, dirigiert der Regent die Dinge und Subjekte. Für dieses Dirigat wurde die Statistik zur Voraussetzung. Dazu bedurfte es neuer medialer Werkzeuge der Informationserhebung und -verarbeitung. Mit der Tabelle im Fokus soll diese Studie der Entwicklung dieser Werkzeuge nachgehen.

In die Schaffung einer ordnenden, etatistischen Sicht auf die Welt reiht sich eine Schrift Gottfried Wilhelm Leibniz' ein, in welcher er im Frühjahr 1680 Überlegungen zum *Entwurf gewisser Staats-Tafeln* anstellt. Er beginnt: »Ich nenne Staats-Tafeln, eine schriftliche kurze Verfaßung des Kerns aller zu der Landes-Regierung gehörigen Nachrichten, so ein gewisses Land insonderheit betreffen, mit solchen Vortheil eingerichtet, daß der Hohe Landes-Herr alles darinn leicht finden[,] was er bey ieder Begebenheit zu betrachten[,] auf einmal übersehen und sich deßen als eines der bequämsten Instrumenten zu einer löblichen Selbst-Regirung bedienen könne« (Leibniz 1680, 341).<sup>130</sup> Bereits im ersten Satz finden wir eines der Hauptmotive von Tabellenwerken: die geraffte, synchrone Übersicht. Leibniz argumentiert weiter, sich nicht auf (mündliche) Berater zu verlassen, sondern dieses »beqvämste Instrument« schriftlich zu fixieren, damit das Wissen jederzeit verfügbar sei – zum Beispiel auch auf Reisen und Feldzügen. In der Folge seien die Staatsarchive und deren »Rubriken und Register also einzurichten, daß sie endlich in diese Staatstafel als in ein Centrum zusammen lauffen« (ebd.). Er hebt hervor, dass nicht allgemeine Nachrichten aufzunehmen seien, sondern vor allem quantifizierbare Informationsobjekte und er verdeutlicht, was in seinem Sinne überhaupt eine Nachricht sei, am Beispiel der Seidenproduktion und -konsumtion: »was in einem Lande für eine Quantität seidene Zeuge oder wüllene Tücher jährlich consumiret, oder verthan werden« (ebd. 342). Dies sei deswegen eine »Nachricht«, weil es auf Fakten beruhe und das Wissen darüber erlaube, die zukünftige Produktion zu steuern – zu regieren. Im Folgenden plädiert er, mit den oben bereits diskutierten Argumenten der Synchronizität des Betrachtens und des Potenzials der Wissensverknüpfung, für die Verwendung der Tabelle: »Alles aber nicht allein leicht zu finden, sondern auch was zusammen gehöret, gleichsam in einen Augenblick zu übersehen, ist ein weit größerer Vortheil, als der ins gemein bey Inventariis anzutreffen, daher ich dieses Werck Staatstafeln nenne, dann das ist das Amt einer Tafel, daß die Connexion der Dinge sich darinn auf einmahl fürstellet, die sonst ohne mühsames Nachsehen nicht zusammen zu bringen« (ebd. 345). Leibniz argumentiert weiter, dass die Tabelle die mathematischen Daten in Modell und Form brächten, wodurch wiederum »alles in die Enge getrieben und augenscheinlich oder handgreiflich gemacht wird« (ebd.). Ebenso wie Karten den Seefahrer, die Buchhaltung den Kaufmann nützten, so könnten die Staats-Tafeln zur Regierungskunst beitragen. Umgekehrt ermögliche das aus den Tafeln generierte Regierungswissen es, die Menschen »gleichsam an der Schnuhr zu haben« (ebd.). Hier zeichnet sich bereits eine Dialektik der Nähe und Ferne ab, die diagrammatischen Werken als Versprechen gemeint ist: Aus der Ferne erlauben sie eine Totalität, einen Überblick über die Gesamtsituation; im Detail ermöglichen sie den Zugriff auf einzelne Informationsobjekte und die Steuerung einzelner

---

<sup>130</sup> Zu besserer Lesbarkeit sind in den Leibniz-Zitaten die Substantive, so nicht durch ihn geschehen, in der Großschreibung korrigiert, die übrige Schreibweise wurde beibehalten.

Prozesse. Damit erfolgt eine Verschiebung von der Regulierung zur Selbst-Regulierung staatlicher Prozesse. Die Tabelle wird zum Werkzeug eines Feedbackmechanismus und schreibt das Feedback ein für alle mal in die Kunst des Regierens ein: Der lange Weg zur Kybernetik des 20. Jahrhunderts nimmt seinen Anfang.

So bildhaft Leibniz' auch sein mag, es bleibt einer von vielen Entwürfen, die heute in seiner gesammelten Korrespondenz zu finden sind. Unmittelbare Verwirklichung fanden seine Vorschläge nicht. Leibniz' Staats-Tafeln können hier einen historischen Vektor beispielhaft illustrieren – die Tendenzen zur Systematisierung, wie sie sich seit Ende des 16. Jahrhundert entwickelten und immer weitere gesellschaftliche Bereiche durchdrangen. Im Zuge des zunehmenden Einsatzes von Statistik als wirtschaftspolitisches Instrument, entstand der Begriff einer auf den Staat bezogenen politischen Ökonomie, die in Deutschland entlang der fürstlichen Camera als Kameralismus bezeichnet wurde.<sup>131</sup>

Es zeigt sich, dass die statistische Erfassung des Staatswesens durch die beginnende Ökonomisierung der Gesellschaft im Zuge des Frühkapitalismus massiv ausgeweitet wurde. Wie dies konkret aussah, gilt es beispielhaft anhand der Luxusgüterproduktion im Frankreich Mitte des 17. Jahrhunderts zu skizzieren. Leibniz hielt sich für fast vier Jahre (1672-1676) in Paris auf. Dieser Zeitraum überschneidet sich mit der durch Jean Baptiste Colbert ab 1661 am Hof Ludwig XIV. betriebenen Revitalisierung der französischen Wirtschaftsleistung, motiviert durch die beabsichtigte Erhöhung der Einnahmen des Hofes.<sup>132</sup> In diesem Rahmen förderte Colbert die statistische Erfassung und Steuerung der Luxusgüterproduktion, die am französischen Hof, neben dem Militär, eine besondere Rolle spielte.<sup>133</sup> Die Leibnizschen Staats-Tafeln standen im Geiste seiner Zeit.

Es waren die textilen Manufakturen, die neben den militärischen Industrien, ab dem Beginn der Alleinherrschaft Ludwig XIV., in das Zentrum staatlichen Planungsinteresses rückten. Dafür spielte der zum Wirtschaftsminister aufgestiegene Kaufmannssohn Jean-Baptiste Colbert (1619–1683), der 1661 an den Hof berufen wurde, eine wesentliche Rolle. Colbert trachtete danach, den Staat zentralistisch regierbar zu machen. Er sorgte für eine weitreichende Institutionalisierung der Kunst- und Luxusgüterproduktion.<sup>134</sup> Damit forcierte er eine Ökonomie, die darauf ausgerichtet war, noch aus dem letzten Winkel des Landes zum Glorifizierungsbestreben des königlichen Hofes beizutragen. Colbert selbst baute seine Dienste für den König auf einer komplexen Wissenssammlung auf, welche 1683, in der königlichen Bibliothek und in seiner nicht minder umfangreichen privaten Bibliothek insgesamt 59.000 gedruckte Bücher und 15.600 Manuskripte umfasste. In der Frühneuzeit repräsentiert Colbert wie kein anderer den Wandel von einer politischen Auffassung des Regierens zu einer technokratischen.

Wie aber konnte es gelingen, die im Land verstreute Produktion zentralistisch zu steuern und zum Ruhme des Königs auf ein neues Produktionsniveau zu bringen? Ein Heer von Inspektoren eilte als Berichterstatter über Land, um die königlichen Befehle durchzusetzen und den Stand der Dinge an den

---

<sup>131</sup> Zum Kameralismus vgl. *Kalkül und Leidenschaft – Poetik des ökonomischen Menschen* (Vogl 2002, 57ff.). Die Verbreitung der *Polizey-Wissenschaft* ist auf das in Frankreich und Deutschland intensiv rezipierte Kompendium *Traité de Police* von Nicolas Delamare zurück zu führen, welches 1705–1719 in drei Bänden erschien.

<sup>132</sup> Leibniz korrespondierte dort u.a. mit dem niederländischen Wissenschaftler und Mathematiker Christiaan Huygens, der auf Betreiben von Colbert als Mitglied der Akademie der Wissenschaften berufen wurde (Nadler 2010, 18).

<sup>133</sup> Vgl. *Die innere französische Gewerbepolitik von Colbert bis Turgot* (Farnam 1878), *La fortune du colbertisme – Etat et industrie dans la France des Lumières* (Minard 1998), *The Information Master – Jean-Baptiste Colbert's secret state intelligence system* (Soll 2009) und *The Reckoning – Financial Accountability and the Rise and Fall of Nations* (Soll 2014, 119–133).

<sup>134</sup> Vgl. *Textiles Prozessieren – eine Mediengeschichte der Lochkartenweberei* (Schneider 2007, 211, Fußnote 6).



Hof zu berichten.<sup>135</sup> Neben beschreibenden Reiseberichten, gliedert entlang der durch die Inspektoren aufgesuchten Orte, schrieb das Colbert unterstehende *Bureau de Commerce* ihnen eine Form vor, welche Übersichtlichkeit, Standardisierung und Vollständigkeit forcierte: die Tabelle. Häufig erfassten die Inspektoren »im Feld« die Daten in Listenform, welche ihrerseits standardisiert war. Auch einheitliche Formulare wurden intern immer wieder diskutiert, jedoch bis zum Ende des *Bureau de Commerce*, 1791, nicht durchgesetzt. Die datentechnische Verarbeitung dieser Listen erfolgte nach einem variablen Schema: In einem ersten Schritt wurden Sachverhalte durch direkte Beobachtung erhoben und in Fragebögen, Listen oder ähnlichem festgehalten. Diese »Urlisten« wurden im Legeverfahren nach verschiedenen Merkmalen sortiert und gruppiert, oder es wurden im Listenverfahren spezifische Informationen gezählt, gelistet und summiert. Das daraus abgeleitete Wissen diente als Grundlage für weitere Entscheidungen.

Den Inspektoren fällt die Aufgabe zu, die Daten zu »erheben«, also im wortwörtlichen Sinne aus dem Alltag und aus dem von ihnen verfassten Fließtext heraus zu heben, sie zu heben, zu erhöhen, gegenüber einer Vielzahl von Tatbeständen. Sie betreiben eine Datenerfassung und auch die Erfassung soll wortwörtlich verstanden werden als ein Vorgang, der dazu führt, etwas zu fassen und in der Konsequenz von seinem ursprünglichen Ort fort zu tragen, während anderes als »Nicht-Daten« liegen blieb.<sup>136</sup> Aller sechs Monate berichteten sie an das *Bureau de Commerce*, welches die Berichte wiederum in Übersichtstabellen dem König zur Verfügung stellte (Abb. 21).

Tableau des recettes et des dépenses depuis 1662 jusques et y compris 1680<sup>1</sup>.

ANNÉES	RECETTE.	DÉPENSE.	EXCÉDANTS	
			DE RECETTE.	DE DÉPENSE.
1662	75,568,750 <sup>1</sup>	74,826,456 <sup>1</sup>	742,294 <sup>1</sup>	"
1663	48,053,826	46,826,576	1,227,250	"
1664	63,002,796	63,071,008	531,788	"
1665	90,883,973	90,871,856	12,117	"
1666	67,459,001	66,611,895	847,106	"
1667	72,520,925	72,090,744	430,181	"
1668	70,875,374	70,875,381	"	7 <sup>1</sup>
1669	76,468,967	76,283,149	185,818	"
1670	73,900,755	77,209,879	"	3,309,124
1671	87,501,077	83,875,723	3,625,354	"
1672	87,067,787	87,928,561	"	860,774
1673	96,971,302	98,242,773	"	1,271,471
1674	105,738,044	106,803,861	"	1,065,817
1675	112,133,054	111,866,488	266,566	"
1676	110,936,796	110,132,622	804,174	"
1677	116,315,294	115,819,462	495,832	"
1678	106,705,242	106,910,519	"	205,277
1679	126,132,816	128,235,300	"	2,102,484
1680	91,759,460	96,318,016	"	4,558,556
TOTAL.....	1,680,595,229	1,684,800,269	9,168,480	13,373,510

Excédant des dépenses sur les recettes pendant la période de 1662 à 1680, inclusivement: 4,205,030 livres.

Abb. 21: Einnahmen (Recette) und Ausgaben (Dépense), sowie Überschüsse (Excédants) der Einnahmen und der Ausgaben. Nachabdruck der ursprünglich (vermutlich) handgeschriebenen Tabelle aus dem Abregé des Königs von 1680. ([1680] Colbert 1865, II/2:783).

<sup>135</sup> Einen Einblick gibt Farnam 1878 in (Minard 1998, 147–261), hier auch Quellen-Verweise auf einzelne Berichte der Inspektoren, zum Beispiel S. 185, Fußnote 1. Vgl. auch die Beschreibung der Aufgaben der Inspektoren in (Soll 2009, 67–69).

<sup>136</sup> Mit Latour ließe sich die Tabelle der erhobenen Daten den unveränderlich mobilen Elemente (Immutable Mobiles) zuschlagen (Latour 2006, 264f., 285f.), welche er als mobil, unveränderlich, verflüchtigt, veränderbar im Maßstab, reproduzierbar, kombinier- und überlagerbar kennzeichnet und deren besonderes Potenzial in ihrer geometrischen Qualität liegt, die erlaubt Raum und Zahl miteinander zu verschmelzen (bei der Tabelle wäre dieser Fall zum Beispiel beobachten, wenn die Zeilen anhand von Zahlen aufsteigend sortiert werden).

Tatsächlich hatte es sich der junge Ludwig XIV. zur Gewohnheit gemacht, zweimal täglich, die von Colbert zur Verfügung gestellten Berichte zu Themen wie Kontenstand, Truppenzahlen, diplomatische Informationen zu studieren. Der König wurde, als erster französischer König überhaupt, durch Colbert anhand der Lehren Luca Paciolis mit der doppelten Buchführung bekannt gemacht (Soll 2009, 60ff.). Die Kunst des Regierens – die Berater der französischen Könige hatten bis Colbert einen humanistischen oder kirchlichen, jedoch keinen merkantilen Hintergrund – wandelt sich zur Technologie des Regierens.

In den Listen und Tabellen des Bureau de Commerce bildet sich ab, wie quantitative als auch qualitative Informationen auf Attribute verteilt werden, voneinander abgegrenzt und doch vereint dargestellt. Sie wurden Voraussetzung dafür, qualitative Informationen anhand von quantitativen Operationen (z.B. Sortierung, Gruppierung, aber auch Summierung, Mittelwert etc.) neu zu ordnen. Wesentlich für die erfolgreiche Verwendung der Tabelle im königlichen Palast waren zwei Modi von Operation:

- die Erhebung der Daten durch Zählen und Messen, sowie mathematische Berechnungen und
- das Betrachten disparater Daten und Er-Denken ihrer Beziehungen untereinander.

Dem Er-Denken beziehungsweise Konstruieren von Beziehungen, geht ein anderer Schritt voraus, der zur Ausgangsmotivation für die Zusammenstellung einer Tabelle zurückführt: Das Er-Denken wird geleitet von Fragestellungen, die so formuliert sein müssen, dass sie dem Übersichts-Charakter der Tabelle entgegen kommen. Dabei spielen kategoriale und quantitative Vergleiche<sup>137</sup> eine große Rolle, zum Beispiel: ›In welcher Region gibt es mehr Webstühle, in der Normandie oder um Lyon?‹, ›In welcher Jahreszeit wird mehr Wolle produziert?‹. Diese Fragestellungen trieben das Informationsbedürfnis an, sie modellierten, welcher Anteil von Realität als Daten erhoben würde und welcher Anteil zu ignorieren war.

Zusammenfassend ist festzuhalten: Während in Zedlers Lexikon von 1744 die Tabelle als Wissensspeicher und Memorierungswerkzeug auftritt (es gibt allerdings auch einen Eintrag für Astronomische Tafeln), hat in den folgenden Jahrhunderten ein Wandel des Verständnisses von Tabellen stattgefunden. Am besten illustriert wird diese Verschiebung hin zum statistischen Werkzeug durch eine Blick in Meyers Großes Konversationslexikon in der Ausgabe von 1908: »Von besonderer Bedeutung ist das Tabellenwesen in der Statistik. Die gesetzmäßig wiederkehrenden Zahlenverhältnisse im Wechsel der Bevölkerung u. sind in dieser Wissenschaft in feste Tafeln gebracht worden, auf denen sich dann die praktischen Schlussfolgerungen aufbauen, wie z.B. die Berechnung der Beiträge für Lebensversicherung, Witwenversorgung u. auf den Sterblichkeitstabellen« (Meyer/Joseph 1908, 274).<sup>138</sup> In diesen Ausführungen wird sichtbar, wie die Tabelle als Instrument der Statistik zur Hervorbringung einer gouvernementalen Regierungsweise genutzt wird, welche die Bevölkerung adressiert: Bevölkerungszahlen,

---

<sup>137</sup> Stephen Few unterteilt die möglichen Beziehungen in folgende Unterpunkte: 1.) Kategoriale Beziehungen: nominal – ordinal/geordnet – Intervall – hierarchisch; 2.) Quantitative Beziehungen: Rangfolge – Verhältnis – Korrelation/ Wechselbeziehung. Vgl. *Show me the Numbers* (Few 2012, 17–22).

<sup>138</sup> Es werden auch andere Tabellenformen aufgezählt, die zum Zwecke des Verständnis aus dem Zitat gekürzt wurden: »Dahin gehören unter anderem Geschichtstabellen, Regenten- und Stammtafeln, tabellarische Übersichten naturhistorischer Systeme, des spezifischen Gewichts der wichtigsten Naturkörper, des Atomgewichts der Elemente; auch Logarithmentafeln, Zins- und Zinseszinstabellen für Arithmetik und Trigonometrie u.a.« (ebd.). Diese Tabellen werden in einem anderen Abschnitt ausführlicher diskutiert.

Berechnungen zur Lebensversicherung sind dafür beredtes Zeugnis. Das hierin generierte Wissen fließt ein in eine Polizeywissenschaft, die auf Wohlergehen und Disziplinierung der Regierten abzielt. Die Staatstabelle übernimmt dabei die Rolle eines grundlegenden Beschreibungssystems des Staates. Sie löst damit den Körper des Regenten ab, der bis dahin als Staatskörper diente.

Das Konzept der Staatstabelle verweist auf die nunmehr entstehenden Formen von Gouvernementalität, einem ›Dreieck‹ aus Souveränität, Disziplin und gouvernementaler Verwaltung. Voraussetzung für die Berechnungen, Analysen und Reflexionen ist das Zusammenführen und Kategorisieren von Informationen in der Tabelle, also das Verorten der Gegenstände in eine symbolische Ordnung. Die Staatstabelle ist somit Medium und Werkzeug des Denkens und wird zum Instrument gouvernementaler Souveränität.<sup>139</sup> Doch können Tabellen noch deutlich intensiver operationalisiert werden, dies wird in den folgenden Beispielen diskutiert.

### 2.2.2.3 Transaktionen – Plantafeln: Prozess- und Organisationstabellen

»(1) route clerks, (2) instruction card clerks, (3) cost and time clerks, [...] (4) gang bosses, (5) speed bosses, (6) inspectors, (7) repair bosses [...] and (8) the shop disciplinarian«  
(Taylor/Frederick Winslow 1903, 104).

Während Staatstabellen der Auswertung des Datums, des Gegebenen dienen, bilden Transaktionstabellen laufende und kommende (Arbeits-)Prozesse und Transaktionen ab. Sie fungieren als interne Verwaltungswerkzeuge der Zuteilung von Ressourcen. Eingeführt wurde diese tabellarische Form im Zuge zunehmender Arbeitsteilung und der, durch Frederick W. Taylor propagierten, Verwissenschaftlichung der Produktion (Taylor/Frederick Winslow 1903, 64f., 102f., 117ff.; Taylor/Frederick Winslow 1911) seit den 1910er Jahren. Ihren großen Auftritt erhält die Plantafel damit in der Übergangsperiode von der manufakturrellen Herstellung zur industriellen Produktion und zwar im Zuge zunehmend notwendiger Informationszirkulation. Neben Produktionsbetrieben waren die hauptsächlichlichen Nutzer Versicherungen (Yates 2005, 22, 25), Banken (Batiz-Lazo/Wood 2002, 3) und Kreditkartenunternehmen (Lauer 2020), die vor allem Geldtransaktionen zu bewältigen hatten. Dienstleister, wie American-Airlines, setzten Plantafeln in den 1930er bis 1980er Jahren für Flugbuchungen und Auslastungsanzeigen ein. Ab den 1960er Jahren wurden sie sukzessive durch elektronische Buchungssoftware, wie SABRE, ersetzt. Auch Bibliotheken führten neben den eigentlichen Stammdaten, dem Verzeichnis der vorhandene Bücher, Transaktionstabellen oder -karteien, in der die Ausleihen vermerkt waren (Wiegand/Davis (Hrsg.) 1994, 142ff.).

Dispatch-Boards, Dispatch Rack, Production Board, Planning Board, Einstecktafel, Organisationstafel und Plantafel waren tabellarische Figurationen, die nicht nur eine räumliche Dimension der Informationen erlaubten, sondern auch eine zusätzliche Farb-Topologie zur Verfügung stellten (Abb. 22). Sie dienten der Beherrschung von Transaktionen, also zeitlich bestimmten Abläufen, die auf einen bestimmten Gegenstand bezogen wurden und ggf. Teilaktionen oder Folgeaktionen nach sich zogen, zum Beispiel die Nutzungsdauer und damit Auslastung einer Maschine zur Herstellung eines Einzelteiles und die Folgeplanung für das Einzelstück, oder die Belegung eines Sitzplatzes im Flugzeug über eine

---

<sup>139</sup> Allerdings kommt die Tabelle als diagrammatisches Mittel der Übersicht schnell an ihre Grenzen. Mit der Zunahme der Datenpunkte, die einem allgemeinen Anstieg der statistischen Datenerhebung folgen, wird der durch die Tabellen eingenommene Datenraum zu ausschweifend, es gilt neue Methoden der bildlichen Verdichtung von Daten zu entwickeln. Dieser Umstand führte zu neuen Qualitäten des Gebrauchs von Karten und Diagrammen.

spezifische Dauer und Strecke mit eventuellen Anschlussflügen, oder die Ausleihe eines Buches über einen bestimmten Zeitraum. Diese Abläufe zeichnen sich durch einen Anfang und ein Ende aus, und es war notwendig, sicherzustellen, dass sie auch tatsächlich erfolgreich beendet worden waren.<sup>140</sup> Ein weiteres Merkmal der Plantafeln war die Abbildung und Simulation zeitlicher Reihenfolgen, z.B. die Umrüst- und Reparaturzeiten von Maschinen im Produktionsvorlauf, oder die Dauer für die eine Maschine jeweils belegt war. Sie »[...] muss alle Plätze zeigen, wo Arbeit durchgeführt werden kann und auch der in Bearbeitung befindliche Auftrag und mindestens einen Auftrag je Maschine oder Arbeitsplatz im Voraus« (Alford/Bangs (Hrsg.) 1944, 188).<sup>lxxxii</sup>

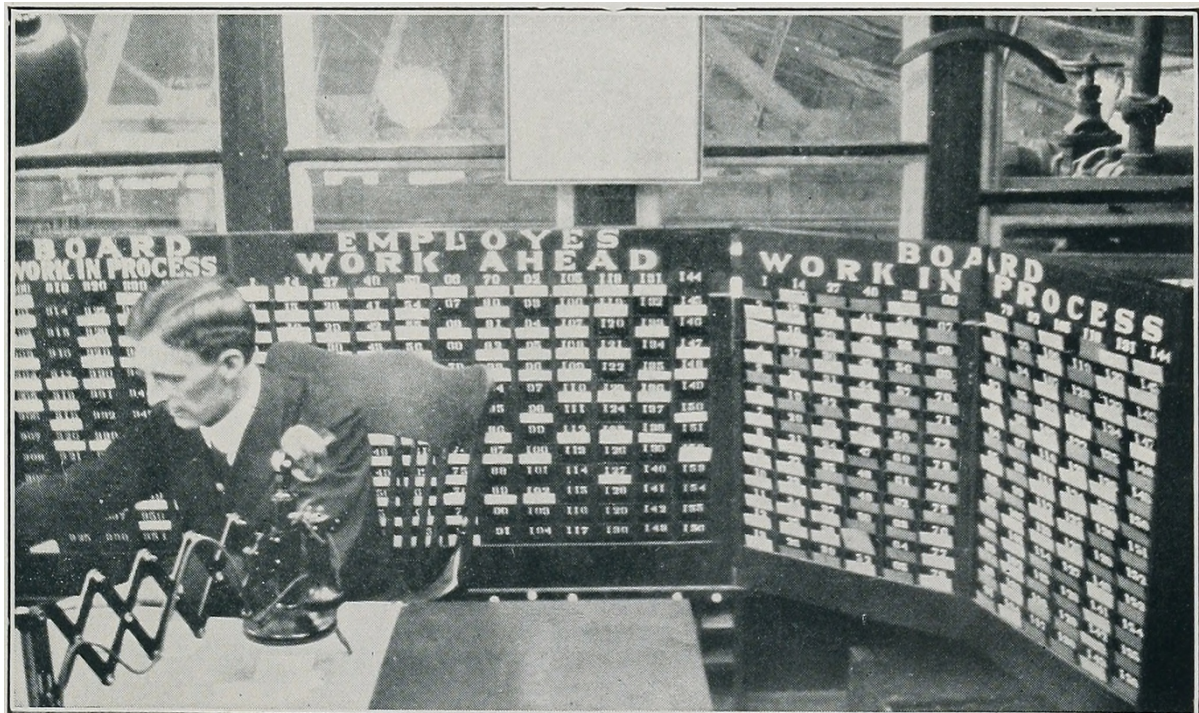


Abb. 22: Dispatch Board: Ein Arbeiter kommt ins Dispatcher-Büro, nennt die Maschinenummer und erhält den nächsten verfügbaren Auftrag für diese Maschine. Der Dispatcher notiert Beginn und Ende jedes Arbeitsvorgangs je Maschinenarbeitsplatz und informiert die Planungsabteilung falls Maschinen nicht genügend ausgelastet sind. Unter ›Work in Process‹ sind die Karten auch farblich codiert. Leerstellen zeigen Leerlauf oder Abwesenheit an (Van Deventer 1921, 235; Abbildung ohne Jahr und Ort, vermutlich 1910–1920er Jahre).

Die tabellarische Form entspringt dabei einem Informationsbedürfnis, welches Zeit als linear und diskret operabel sinnlich zugänglich macht. Ein historisches Farbsystem, von dem Van Deventer 1921 in *Planning Production for Profit* berichtete, diente der Signalisierung der Zustände, insbesondere von Leerlauf: »Leerlauf wird farblich wie folgt hervorgehoben: Arbeit, rot – Material, schwarz – Abwesenheit, gelb – Werkzeuge, blau – Maschine, grün« (Van Deventer 1921, 274).<sup>lxxxiii</sup> Karten der jeweiligen Farbe waren für ein bestimmtes Datum (Spalte) und eine bestimmte Maschine (Zeile) einzusortieren. Aldford und Bangs diskutierten 1944 in *Production Handbook* eine Plantafel, in der grüne und gelbe

<sup>140</sup> Um die Darstellung konzentrieren zu können, wird darauf verzichtet, eine besondere Form der Plantabelle, die sogenannten Gantt-Charts, auszuführen. Diese betonten zusätzlich zur zeitlichen Dimension die Pfadabhängigkeiten von Prozessen. Gantts Artikel zur Zeitplanung war 1903 ursprünglich parallel zur Veröffentlichung Frederick W. Taylors *Shop Management* erschienen, und integral mit Taylors Ansatz angelegt (Wilson/James M. 2003, 430). In der Folge wurden Gantt-Charts weiter entwickelt und erschienen in zahlreichen Büchern zur Produktionsplanung (z.B. in: Clark 1922; Alford/Bangs (Hrsg.) 1944, 216–229), wobei sie als relativ komplex in der Handhabung eingeschätzt wurden (Herrmann 2006, 5–10).

Zettel-Formulare einem Tag (Spalte) und einem Maschinenarbeitsplatz (Zeile) zugeordnet sind. Während Grün den Beginn von Arbeiten und der damit verbundenen Informations- und Organisationsanforderungen signalisiert, steht gelb für deren Fortgang. Blaue Zettel signalisieren den Abschluss von Arbeiten und Pink zeigen Probleme und Verzögerungen an. Damit kann Tag für Tag der Fortgang der Arbeiten je Maschine visualisiert werden (Alford/Bangs (Hrsg.) 1944, 196).

Der Farbgebung kam, wie zu sehen ist, vor allem Signalwirkung zu, denn ebenso wie eine Leerstelle zeigte sie beim Tabellieren Aktionspotenziale an. Farbliche Codierung unterstützte die Überblicksfunktion von Tabellen und das visuelle Eingrenzen von Fragestellungen, die sich auf bestimmte Aktionen orientierten (z.B. Wo könnte Reparatur nötig sein?). Hierin tritt erneut das Potenzial aus Distanz (Übersicht) und Nähe (Detail) der Tabelle zum Vorschein.

Die farbige Karten konnten mit weiteren Informationen beschriftet werden, z.B. einer Angabe von Gründen (Krankheit, oder fehlende Materialien), um die notwendigen oder möglichen Planungsaktionen vorzustrukturieren. Ein häufiger Operationsmodus war das Vermerken von Bestellnummern auf den Karten, anhand derer ein Formular mit zugehörigen Informationen aus Karteikästen oder Ordnern entnommen werden konnte, um weitere Informationen zu erhalten (Abb. 23). Die besondere Stärke dieser Figuration liegt in der Verknüpfung der temporalen und topologischen Ebene, welche Operationen nicht nur im Tabellenraum ermöglichen, sondern Operationen im Tabellenraum als Manipulation von Wirklichkeit evozieren. Daraus formt sich die Frage, warum sich die Informationsbedürfnisse in der Plantafel auf diese spezifische Weise ausdrückten?

Drei Funktionen sollten die Plantafeln in Produktionsbetrieben erfüllen: Sie dienten der Planung von Kapazitäten, Materialien und der Produktionsvorbereitung, der Arbeitsabfolge, die vorab optimal entschieden werden sollte und der Zeitplanung, welche die Belegungszeiten einer Maschine, Beginn und Ende eines Produktionsschritts und Problemmanagement, sowie Feedback an Kunden zum Fertigungsstand ermöglichte (Van Deventer 1921, 228f.). Formulare, Tabellen und Transaktionstafeln griffen anhand eines spezifischen Merkmals ineinander, häufig anhand der fortlaufenden Bestellnummer, welche als eindeutiger Schlüssel, bzw. als Adresse des Vorgangs diente. An verschiedenen Orten in einem Betrieb oder in einer Organisation waren jeweils unterschiedliche, spezifische Informationsbedürfnisse zu erfüllen. Wichtig war, diese miteinander zu synchronisieren und zu verweben. So befanden sich an den einzelnen Maschinenarbeitsplätzen Informationen zur Belegung der jeweiligen Maschine, oder der Beginn eines neuen Produktionsschrittes zog nach sich, dass eine Materialliste, die Bills of Material, abgeglichen werden musste und im Lager Material entnommen wurde, ein Vorgang, der wiederum im Zuge der Lagerhaltung zu verzeichnen war. Hierfür wurden Verknüpfungen realisiert, indem Teilenummer, Maschinenummer, Bestellnummer als Adressen angegeben wurden, unter denen die Informationen gespeichert waren.

Das Ineinandergreifen verschiedenster Informationsmittel, die sich durch ihre tabulare Form auszeichneten, spannte ein Netzwerk von Informationsrouten durch die organisatorischen Einheiten. Das Informationsnetz war zu seinem Funktionieren darauf angewiesen, dass nicht nur die zugrundeliegenden oder beabsichtigten Aktionen durchgeführt wurden, sondern dass auch die Informationen über die Vorgänge an genau dem richtigen Knotenpunkt aktualisiert und dann weiter getragen wurden. Ein Augenmerk bei der Produktionsplanung lag daher auch auf der Frage, wer, wann und wie die Informationen übertrug und aktualisierte. Taylors imaginiertes Planungsbüro zeichnete sich durch die Innovation aus, dass diese Delegationen jeweils einzelnen Route Clerks, Instruction Card Clerks und Cost and

Time Clerks zugeordnet wurden (Taylor/Frederick Winslow 1903, 114).<sup>141</sup> In den Clerks kommt aus der heutigen Perspektive die Möglichkeit der Automatisierung zum Ausdruck, indem die Personen eingespart wurden und stattdessen Protokolle, wie das Hypertext-Transfer-Protocol <http://> oder Informationsfigurationen wie relationale Datenbankmanagementsysteme diese Aufgaben übernahmen. Bis in die 1970er Jahre hin, war dies jedoch eine Wunschvorstellung, und wir werden in Kürze sehen, dass Clerks auch heute noch eine höchst wichtige Rolle spielen.

REGULAR CONTINUATION REOPERATION					DISPATCH ORDER MEMORANDUM		L.V.E. FORM 104 PART NO. 1155	
DELIVERIES					DEFECTIVES		ORDER NO. 22386	LOT NO. 1
DATE	QUANTITY	DATE	QUANTITY	DATE	QUANTITY	TO START 12-20-'18	STARTED 8 AM	
						EST. DIE TIME	ACTUAL TIME	
						EST. SET-UP TIME	ACTUAL OPN TIME	
OPN No.	DEPT. No.	MACH. No.	CLOCK No.	OPERATION				3.5 hrs.
2	6	D-26		Milling				
PART NAME					Finion Bolt			SET-UP TIME EFF %
PIECES THIS LOT					PIECES MADE THIS OPERATION		BAL. TO BE MADE THIS OPERATION	
1200							OPN TIME EFF %	
							TOTAL EFF O-DT %	

REGULAR CONTINUATION RE-OPERATION					DISPATCH ORDER TIME CARD		L.V.E. 104-B PART NO. 1155	
							ORD No. 22386	LOT No. 1
							TO START 12-20-'18	BURDEN No.
							EST. OPR. TIME 3.5 hrs	BURDEN RATE
							EST. SET-UP TIME	BURDEN AMT.
OPN No.	DEPT. No.	MACH. No.	CLOCK No.	OPERATION				HOURS
2	6	D-26		Milling				RATE
PART NAME					Finion Bolt			AMOUNT
PIECES THIS LOT					PIECES MADE TO-DAY THIS OPERATION		BAL. TO BE OPERATED THIS LOT	
1200								

Abb. 23: Produktionslaufkarte – Diese Karte identifiziert den Vorgang anhand der Bestellnummer, außerdem die Teilenummer, das geplante Produktionsdatum, die Startzeit und ggf. die Zeit zur Maschineneinrichtung, sowie die geforderte Stückzahl. Sie lag oder steckte in einer Warteschlange der abzuarbeitenden Aufträge und wurden von der Vorbereitungs- in die Auftragschlange verlagert, wenn alle notwendigen Vorbedingungen (Vorhandensein der benötigten Einzelteile und Informationen) gegeben waren. Dann konnte auch der Status in der Plantafel geändert werden (Van Deventer 1921, 233ff.).

<sup>141</sup> Diese Funktionen können sich auch in einer Person bündeln, so in Frau J., welche Lisa Conrad im Betrieb begleitete und interviewte: »Mindestens einmal am Tag geht sie ins Schichtführerbüro und steckt pinke und weiße Kärtchen an die für sie vorgesehenen Positionen: »Wenn es an der Zeit ist, bringen wir immer laufend Aufträge in die Produktion, also nicht das ganze Jahr im Voraus, sondern die feingeplanten: eine Woche oder zwei Wochen.« (Conrad 2017, 84f.). Das Betreten der Produktions-halle zieht einen Rücklauf der Informationen nach sich, da Frau J. dort Erkundigen einzieht, auch implizites Wissen, z.B. über die am besten geeigneten Reihenfolgen von Aufträgen – Berufserfahrung – akkumuliert. Die Rolle der Clerks ist daher nicht unterzubewerten, als Maintainer sorgen sie für den Informationsfluss, ohne den die Produktion stoppen würde.

In den Formularen fällt ein Unterschied in den einzutragenden Daten ins Auge, der sich in den Strukturen der tabellarischen Darstellung spiegelt. Ein Teil der Informationen im betrieblichen Ökosystem waren eher statischer Natur, zum Beispiel all jene, die sich unter Stammdaten fassen ließen. Dazu zählen Informationen zu einem Mitarbeiter (Name, Adresse, Kontoverbindung etc.), die Einzelteilliste eines Produktes, die Informationen zu einer einzelnen Maschine (Fähigkeiten der Maschine, Baujahr, Verschleißzustand), oder die Informationen zu einem Buch (Autor, Titel, Verlag, etc.). Dies sind Informationen topologischer Natur. Diese Daten, einmal eingetragen, änderten sich nur in geringem Maße und waren daher von hoher Dauerhaftigkeit.

Im Unterschied dazu und im Unterschied zu den anderen hier besprochenen Tabellenarten (Wissenstabelle, Staatstabelle, Mathematische Tabelle), ist die Transaktionstabelle hoch dynamisch. Sie spiegelt im Idealfall zu jedem Zeitpunkt den aktuellsten und immer wieder aktualisierten Stand der Transaktionen eines organisatorischen Zusammenhanges wieder. Sie zielt auf die Gebrauchs- und Handlungsdimension ab, und damit einhergehend auf verschiedene Zustände, welche von Gegenständen oder Personen durchlaufen werden. Dieser Unterschied ist ein Unterschied ums Ganze – erst dadurch wird die Transaktionstabelle unterscheidbar.

Die Vermutung mag naheliegen, dass die Transaktionstabelle heute komplett automatisiert wurde und beispielweise Element von Enterprise-Ressource-Planning Figurationen geworden ist. Bezeichnenderweise ist sie nach wie vor in heutigen Betrieben und Organisationen anzutreffen und wurde durch Lisa Conrads als *Plantafel*, bzw. als »bewegliche Tabelle« eindrücklich dargestellt.<sup>142</sup> Ihre Nutzung erscheint den Beteiligten opaque: »Von ihr geht keine wahrgenommene Intervention aus. Sie verhält sich die meiste Zeit als getreues Zwischenglied und wird die meiste Zeit auch als solches behandelt. Das Arrangement ist so erfolgreich, dass es sich unsichtbar macht« (Conrad 2017, 79). Es falle der Plantafel schwer, ihren eigenen Wert zu behaupten, denn sie wirke aus Sicht der Mitarbeiter antiquiert und unmodern (ebd. 89). Jedoch, so stellt Conrad heraus, präsentiere sich die Plantafel als un abgeschlossenes Ding, welches in ständigen Beziehungen zu Personen und Formularen steht und Infrastruktur geworden ist.

Das legt die Vermutung nahe, dass gerade die visuelle Anmutung, die Möglichkeit, Überblick herzustellen und sie öffentlich in einem Raum, an die Wand montiert, auszustellen, eine wesentliche Qualität der Plantafel ist. »Einen einzelnen Arbeitstag zu fokussieren geht einher damit, diesen zu einer größeren Gruppe oder der Gesamtheit von Arbeitstagen in Beziehung zu setzen. Auf diese Weise produziert die Plantafel weitere tabellenspezifische Aussagen (vgl. Gregory 2013:10). Sie generiert Aussagen über die Gesamtsituation der Produktion, die sich in Verdichtungen von pinken Karten (hohe Auslastung) auf der einen Seite und Flächen aus grünen Karten (geringe Auslastung) auf der anderen Seite ausdrückt« (Conrad 2017, 72). Diese Art der Operationalisierung muss nicht für jede Transaktionstabelle der Fall sein. So können beispielsweise das Verzeichnen von Transaktionen in Banken oder von Nutzeraktionen in Web 2.0-Plattformen im Bigtable-Format weitestgehend unsichtbar bleiben und allein der maschinellen, automatisierten Verzeichnung, Manipulation und Auswertung dienen.

Die spezifische Möglichkeit der Plantafel besteht im Unterschied dazu im gemeinsamen Bearbeiten von Informationen, in der gemeinsamen Sichtung durch Personen, die sich in einem Raum aufhalten (ebd., 69). Darin liegt auch einer der Gründe, dass sie bisher nicht vollständig in ERP-Systemen aufgegangen ist, auch wenn entsprechende Module, beispielsweise in SAP, im Prinzip vorliegen. Im Potenzial

---

<sup>142</sup> Heute werden Plantafeln in Deutschland beispielsweise unter der Bezeichnung Einstecktafel, und in Großbritannien als *T-Card Maintenance Board* vertrieben.

der der impliziten Informationsweitergabe und -verarbeitung durch die Clerks liegt ein weiter Grund für die Persistenz des Prinzips Plantafel: Dadurch, dass in vollautomatisierten Figurationen jegliche Information formalisiert werden muss, um existent zu sein, kommen diese an ihre Grenzen, wann immer die Situation informelle, nicht-formalisierte Informationsflüsse erfordert. In Grenzsituationen sind es die informellen Informationsbedürfnisse, die sich mit gewachsenem Erfahrungswissen der Clerks kombinieren und auf diesem Wege das Gelingen von Transaktionen ermöglichen.

Zusammenfassend können Transaktionstabellen als prozessorientierte Verknüpfungstopologien statischer und dynamischer, das heißt zeitabhängiger Informationen gekennzeichnet werden. Sie ermöglichen Operationsketteten und organisieren diese. Die Plantafel als Sonderfall der Transaktionstabelle dient der Standardisierung von Aktionen. Sie werden einerseits abgebildet, um Übersicht zu schaffen, andererseits werden Aktionen mit Hilfe farbiger Codierung geplant, durchgeführt und abgeschlossen. Transaktionstabellen stehen nicht für sich allein, sondern sind Teil komplexer tabellarischer Verweisfigurationen, die es erlauben, dynamische und statische Daten in Beziehung zu setzen und Information örtlich zu verteilen und kontextuell abhängig zu verarbeiten. Fortlaufende Bestellnummern dienten dabei als Adressen, unter denen Vorgänge dauerhaft verzeichnet wurden und prinzipiell jederzeit erneut aufgefunden werden konnten.

Die Plantafel ist dabei zentral im Planungsbüro als Infrastruktur gesetzt. Weniger spektakulär als im von Latour gewählten Beispiel des Seefahrers Jean-François de La Pérouse, ergeben die Inskriptionen der mit der Transaktionstabelle verknüpften Aktions-Formulare nur Sinn, wenn diese zurückkehren in das Planungsbüro. Aus dem Zurückkehren der Inskriptionen entspringt die Macht des Planungsbüros (vgl. Latour 2006, 265; Conrad 2017, 78).

Transaktionstabellen und Plantafeln gaben in den 1960er Jahre Anstoß für computerisierte Manufacturing Resource Planning (MRP) Software, welche wiederum im Zusammenspiel mit Datenbanksoftware stand. Diese Entwicklungen sind konzeptuell im Abschnitt 2.3. *Koordination – Transaktionen* ausgeführt. Der Abschnitt 3.2 *Robotron und SAP: Sozialistisches und kapitalistisches Enterprise Resource Management im Vergleich* vertieft diese Auseinandersetzung am Fallbeispiel von MRP-Systemen in der DDR. Vorerst aber soll die Mathematische Tabelle näher gekennzeichnet werden, um das kategoriale Anliegen des vorliegenden Kapitels abzurunden.

#### 2.2.2.4 Mathematische Tabellen – Menschliche Rechner, Logarithmen und die Massenproduktion mathematischer Ergebnisse

»Falls das Koordinatengitter 200 km im Quadrat für die Ebene umfasste, gäbe es 3.200 Spalten für die gesamte Karte des Globus. In den Tropen ist das Wetter häufig voraussehbar, so dass wir von 2.000 aktiven Spalten ausgehen können. Es würden also  $32 \times 2.000 = 64.000$  Rechner benötigt, das Wetter für den ganze Globus vorher zusagen« (Richardson 1922, 122).<sup>lxxxiv</sup>

Die Vielzahl von Typen mathematischer Tabellen und spricht von einer enormen Bandbreite und darin lassen sich zwei grundlegende Formen unterscheiden:



- Erstens ist die Informationssammlung in abgeschlossener, finaler Form basierend auf einer empirischen Datengrundlage, wie zum Beispiel Fahrpläne oder Ergebnisse physikalischer Experimente beobachtbar.
- Zweitens treten theoretische Zahlenreihen als Hilfsmittel für weitere Berechnungen, wie zum Beispiel Quadratzahlen, Logarithmen, trigonometrische Funktionen, Zinsberechnungen auf (Abb. 24).

Zwischen diesen beiden Polen Empirie und Berechnung finden sich Mischformen, wie die Bahnrechnungen astronomischer Körper, welche von empirischen Daten ausgehend mit berechneten Daten angereichert wurden, zum Beispiel die Flugbahn und Wiederkehr des Halleyschen Kometen.

( 224 )

## TABLES

OF

## INTEREST

*At Five per Cent. per Annum.*

To find the Number of Days from one Month to another,  
see the Table at the End.

1 DAY.		2 DAYS.		3 DAYS.		4 DAYS.		5 DAYS.			
l.	s.	d.	q.	l.	s.	d.	q.	l.	s.	d.	q.
1	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0
40	0	0	0	0	0	0	0	0	0	0	0
50	0	0	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0
70	0	0	0	0	0	0	0	0	0	0	0
80	0	0	0	0	0	0	0	0	0	0	0
90	0	0	0	0	0	0	0	0	0	0	0
100	0	0	0	0	0	0	0	0	0	0	0
200	0	0	0	0	0	0	0	0	0	0	0
300	0	0	0	0	0	0	0	0	0	0	0
400	0	0	0	0	0	0	0	0	0	0	0
500	0	0	0	0	0	0	0	0	0	0	0

at 5 per Cent. per Annum. 225

6 DAYS.				7 DAYS.				8 DAYS.				9 DAYS.				10 DAYS.			
l.	s.	d.	q.	l.	s.	d.	q.	l.	s.	d.	q.	l.	s.	d.	q.	l.	s.	d.	q.
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
50	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
70	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
90	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
200	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
300	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
400	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
500	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Abb. 24: Jahreszinsen zu 5 Prozent. Zur Umrechnung der Spaltenbeschriftungen: 4 farthings (q) entsprechen 1 penny (d); 12d entsprechen 1 shilling (s); 20s entsprechen 1 pound (l) (Leybourne 1789, 224f.).<sup>143</sup>

Das mathematische Tabellieren erfuhr mit dem Aufkommen elektronischer Rechner einen Bruch, denn fortan wurden Multiplikations-, Quadratzahl-, Zins- oder Logarithmentabellen immer weniger benötigt, da die Ergebnisse in der Rechenmaschine zur Laufzeit erzeugt werden können. Bis in die

<sup>143</sup> Siehe auch *The Ready Reckoner* (Williams/Johnson 2005).

1950er spielten mathematische Tabellen jedoch eine wichtige Rolle als Wissensspeicher und Hilfswerkzeug angewandter Mathematik.<sup>144</sup> Ihre Herstellung erforderte eine arbeitsteilige Form der mathematischen Produktion, um den ständig steigenden Bedarf an Berechnungen, einhergehend mit der Industrialisierung, zu decken.

Um zu beleuchten, wie einerseits die organisatorische, arbeitsteilige Form des Tabellierens durch rechnende Menschen, die *Human Computer*, in die Logik früher elektronischer Rechenmaschinen einfließt und andererseits die Tabellenform eine der Grundlagen relationaler Datenbanken werden kann, sollen im Folgenden Praktiken der Tabellenproduktion untersucht werden, die umfangreicher Rechenarbeiten bedürfen. Kennzeichnend war die Zunahme von Arbeitsteilung, welche in ›verwissenschaftliche‹, das heißt quantifizierte, tayloristische Praktiken der Arbeitsteilung mündeten.<sup>145</sup>

Auf welche Weise die Tabelle als Denkwerkzeug dem Rechnen Materialität verlieh, soll im Folgenden diskutiert werden. An zwei historischen Beispielen wird erarbeitet, wie Arbeitsteilung und tayloristische Optimierung der Arbeitsabläufe die Tabelle als Medium einsetzten und von der Tabelle geformt wurden. Die Tabelle findet sich im Zentrum einer geschichtlichen Bewegung wieder, in der die Arbeitsteilung der Human Computer sukzessive in maschinelle Abläufe überführt wurde. Wir begegnen dabei verschiedenen Arbeitsformen der Tabellenproduktion, welche in der folgenden Aufzählung keinesfalls eine zeitliche Sukzession kennzeichnen, sondern häufig parallel nachweisbar sind:

- die vereinzelte Tabellenproduzentin, wie zum Beispiel Emma Gifford, die 1914, nach mehreren Jahren Rechenarbeit, fünfhundert Seiten mit Sinusfunktionen veröffentlichte,
- eine gemeinschaftlichen Tabellenproduktion, wie für den frühen Nautical Almanach, der von 1767 an nautische Daten und Berechnungen publizierte,
- die manuellen Rechenbüros, in denen Rechendienstleistungen für unterschiedlichste Aufträge abgeleistet wurden, wie zum Beispiel das Mathematical Table Project im New York der 1930er, mit Berechnungen für das LORAN Navigationssystem der US-Marine,
- daran anschließend teilmechanisierte Rechenbüros, z.B. durch den Einsatz der Mercedes-Euklid Rechenmaschine für die Flugbahn des Sputnik-Satelliten 1957 in der Sowjetunion.
- Schließlich zählt dazu auch der sukzessive globale Einsatz elektronischer Rechner und Rechenmaschinen.

Aus den möglichen Beispielen wurden zwei ausgewählt, in denen die tabellarische Form besonders eng mit den Arbeitsabläufen und deren Optimierung unter dem Aspekt der Arbeitsteilung verbunden sind. Für das erste Fallbeispiel kehren wir zurück nach Frankreich, nicht zu jenem Ludwig des XIV. und seines Finanzministers Colbert, sondern, 60 Jahre später, zu jenem, in welchem nach der Austreibung des Ancien die Republik ausgerufen wurde. 1790 im ersten Jahr nach Beginn der Revolution übernahm Gaspard de Prony (1755–1839) in Paris die Leitung des Katasteramts, des *Bureau du Cadastre*. Eine der

---

<sup>144</sup> So gab es in der Encyclopedia Britannica 1911 keinen Eintrag zu Tabellen an sich, sondern zur mathematischen Tabelle, die als allgemein gesetzt wurde.

<sup>145</sup> Claus Pias leistet eine medientheoretische Auseinandersetzung mit den Arbeitstheorien Frederick W. Taylors in *Computer Spiel Welten* (Pias 2000, 23–43). Während seine Argumentation darauf abzielte, Computerspiele als kybernetische Normierungsapparatur zu kennzeichnen, zielt die vorliegende Studie auf Tabellenpraktiken und -praktiken ab und kommt daher zu anderen Schlüssen. Klassische Studien sind *The Five Dollar Day: Labor Management and Social Control in the Ford Motor Company, 1908- 1921* (Meyer/Stephen 1981) und *From the American System to Mass Production, 1800-1932* (Hounshell 1984). Eine marxistische Untersuchung und Kritik am Taylorismus lieferte *Labor and Monopoly Capital – The Degradation of Work in the Twentieth Century*. (Braverman 1974) und eine aktualisierte operaistische Kritik *Immaterielle Arbeit – Ästhetisierung der Politik und der Produktion unter den Bedingungen des Postfordismus* (Lazzarato 1998).

Aufgaben des Ingenieurs ergaben sich aus einer Entscheidung der Nationalversammlung vom März 1770, für alle Provinzen das Dezimalsystem einzuführen. Er möge, beschied ihm die Akademie der Wissenschaften, neben anderen Berechnungen, auch die Winkelmaße in das Dezimalsystem überführen, das Sexagesimalsystem sei nicht mehr zeitgemäß. Dies betraf vor allem die Schiffsnavigation und architektonisch-konstruktive Berechnungen. Vor de Prony stand die umfangreiche Aufgabe, Sinus-, Cosinus- und andere trigonometrische Tabellen neu zu berechnen.

Die Verleger William Strahan und Thomas Cadell notierten als Adresse in London *In the Strand*, jene Straße, welche die City of London und Westminster miteinander verbindet. Im Jahr 1776 veröffentlichten sie das Buch eines »ehemaligen Professors für Moralphilosophie an der Universität von Glasgow.« Dessen Werk in zwei Bänden, *Wohlstand der Nationen – Eine Untersuchung seiner Natur und seiner Ursachen*, wurde zu einem globalen Erfolg.<sup>146</sup> Gleich auf den ersten Seiten erklärt Adam Smith, dass der Produktivitätsanstieg vor allem einem Faktor zu verdanken sei: der Einführung der Arbeitsteilung. Nicht länger würde eine Person als Einzelhandwerker sukzessive alle notwendigen Schritte zur Fertigstellung eines Gegenstandes ausführen. Vielmehr würden diese Schritte auf mehrere Personen verteilt werden, die jeweils nur noch einen einzelnen Schritt ausführten und im Vergleich eine wesentlich größere Anzahl von Waren – in seinem ersten Beispiel sind es Stecknadeln – herstellten. Der Kern der Analyse Smiths sind

- erstens, Spezialisierung durch Wiederholung,
- zweitens, Zeitersparnis durch Wegfall von Leerlauf und Umstellung auf verschiedene Arbeitsmethoden und
- drittens, Produktivitätsgewinn durch den Einsatz von Maschinen.

Diese Aufzählung lässt sich nicht anders als eine Empfehlung zugunsten von Arbeitsteilung lesen, und fällt dem geplagten Ingenieur de Prony angesichts seiner ausstehenden dezimalen Trigonometrie eindrucklich ins Auge (Grier 2005, 36). Bei Adam Smith kann de Prony lesen, dass sich Arbeitsteilung auch für die Geistesarbeit lohne (Smith 1776, 12f.).<sup>lxxxv</sup> Er sah sich darin bestärkt, sein mathematisches Problem mit industriellen Methoden anzugehen und schreibt rückblickend in seinen Notizen zu den *Grandes Tables*: »Ich stolperte über das Kapitel, in dem sich der Autor mit der Arbeitsteilung auseinandersetzt und als ein Beispiel für die großen Vorteile dieser Methode die Herstellung von Stecknadeln angibt. Plötzlich kam mir die Idee, diese Methode auf die immense Aufgabe, mit der ich belastet wurde, anzuwenden und Logarithmen herzustellen, so wie man Stecknadeln produziert« (de Prony 1824, 4 zit. nach (Grattan-Guinness 2003, 108)).<sup>147</sup> <sup>lxxxvi</sup> Von der industriellen Metapher inspiriert, ging de Prony von kleinen informellen Rechenteams zur bürokratischen Rechenorganisationen über: »Das Bureau de Cadastre war in jeglicher Hinsicht eine ›Fabrik‹ des Rechnens.« (Campbell-Kelly u. a. (Hrsg.) 2003, 10f.).<sup>lxxxvii</sup>

Diese ›Fabrik‹ war in zwei Sektionen unterteilt: Eine Planungseinheit, die die Verteilung der Kopfarbeit organisierte und eine Produktionseinheit, welche die Ergebnisse produzierte. Die Planungseinheit

---

<sup>146</sup> Es erschien bereits ab 1780 in Paris im *Journal de l'agriculture, des arts et du commerce et des finances*, in deutscher Übersetzung 1796 bei Unger in Berlin und in vielen anderen Sprachen.

<sup>147</sup> Die Methoden de Pronys werden hier diskutiert, da er klaren Bezug auf Smiths Beschreibung der Arbeitsteilung nahm und das Prinzip gezielt einsetzte. Diese Darstellung von de Pronys Methoden soll jedoch nicht im Sinne einer Genese, die mit ihm beginnt, verstanden werden. So übernahm beispielsweise Nevil Maskeline bereits vor de Prony Methoden der Arbeitsteilung für die Erstellung des Nautical Almanac ab 1765 (Grier 2005, 30ff.). Methoden der Arbeitsteilung entwickelten sich vielerorts parallel und unabhängig voneinander, auch bereits bevor sie von Smith beschrieben wurden.

bestand einerseits aus temporär beteiligten Beratern, die den wissenschaftlichen Kontext des Vorhabens herstellten: Die Mathematiker Adrien-Marie Legendre (1753–1833) und Lazare-Nicolas-Marquerite Carnot (1753 –1823) erstellten die Formeln für die Berechnungen und untersuchten potenzielle Probleme anhand vorhandener trigonometrischer Tabellen. Ihre Aufgaben lagen im Bereich der theoretischen Mathematik, sie untersuchten die gegebenen Fragestellungen in Bezug auf ihre Mathematisierbarkeit. Acht weitere Mitarbeiter der Planungseinheit, angewandte Mathematiker, organisierten den täglichen Fortgang der Arbeiten. Sie übernahmen Legendres und Carnots trigonometrische Funktionen und überführten sie in eine basale Form: Addition und Subtraktion.<sup>148</sup> Für Multiplikation, Division oder ähnliches waren die unausgebildeten Arbeitskräfte, die man anheuerte, nicht vorgesehen. Die Planer erstellten tabellarische Arbeitsblätter, auf denen jeder zerlegte Rechenschritt aufgeführt und Leerstellen für die Ergebnisse markiert waren und zwar, indem »sie die Spalten der benötigten Werte und die erste mit Werten ausgefüllte Zeile aufschrieben und die Anweisungen für die Herstellung der verbleibenden Einträge auf dem Blatt notierten« (Grattan-Guiness 2003, 109).<sup>lxxxviii</sup>

The image shows two pages of a logarithmic table. The left page is titled 'N. 1 - 200 L. 00 - 30' and the right page is titled 'L. 30 - 60 N. 200 - 400'. Both pages contain columns for 'N' (Number), 'Log' (Logarithm), 'D' (Difference), and 'P. P.' (Page/Part). The table is organized into sections for numbers 1-200 and 200-400, with sub-sections for 100-300 and 300-600. Each section contains multiple columns of data with headers N, Log, D, and P. P.

Abb. 25: Einfache Logarithmen mit fünf Dezimalstellen, basierend auf de Pronys Manuskripten, veröffentlicht fast 100 Jahre später (Logarithmes Vulgaires a Cinq Décimales des Nombres Entiers de 1 a 12.000, in: Service Geographique de l'Armee (De Prony) 1889)

Hier wird die Tabelle als Formular eingesetzt, welches der prozessualen Logik des Füllens von Leerräumen unterliegt. Sie ist Werkzeug der Zerlegung, der Berechnung und sukzessiven Aggregation der

<sup>148</sup> Zur Mathematik de Pronys, siehe *The Computation Factory - de Prony's Project for making Tables in the 1790's* (Grattan-Guiness 2003, 115).

Ergebnisse. Der Planungseinheit oblag es auch, mithilfe der, von den theoretischen Mathematikern erstellten, Formeln die einzelnen Blätter auf ihre Richtigkeit zu überprüfen und Fehlerkorrektur zu betreiben.<sup>149</sup> Die Rechenarbeit wurden zweifach und unabhängig voneinander, mittels verschiedener Formeln, ausgeführt, sodass die Tabellen wechselseitig verglichen werden konnten. Die Produktionseinheit befand sich nicht in einer Fabrikhalle, sondern die Rechenarbeiter verteilen sich über die ganze Stadt. Die 60 bis 80 Rechner arbeiteten die Aufträge zu Hause ab, bevor sie in de Pronys Büro gesammelt wurden und produzierten um die siebenhundert Ergebnisse pro Tag (Grattan-Guinness 2003, 109). Aus Sicht der »Fließbandarbeiter in einer Tabellenfabrik« (ebd.) muss es sich um eine stupide Tätigkeit, ohne Möglichkeiten zur Qualifizierung gehandelt haben. Die ständige Wiederholung der Arbeitsschritte und damit einhergehende Verbesserung der spezifischen Fertigkeiten und die Ersparnis des Wechsels von Arbeitswerkzeugen und -methoden, können als Einlösung der Smithschen Beobachtungen gesehen werden. Sie fanden im rigiden Gitter der Tabelle ihre mediale Entsprechung.

Das Tabellieren als ›Fließband‹, bzw. Operationsverkettung ist besser noch zu einer anderen Periode zu beobachten, den 1920er bis 1950er Jahren. Es soll in einem zweiten Fallbeispiels diskutiert werden.<sup>150</sup> Im Jahr 1921 erschien in der noch in Frakturschrift gesetzten Betriebszeitung der Optikwerke Carl-Zeiss Jena, auf den Seiten 126–128 der Bericht *Die Arbeit unserer Rechner im Betriebe*. Dr. August Sonnefeld (Abb. 26), Leiter der Optik-Rechner, stellte darin die Tätigkeiten seiner Abteilung dar und erläuterte Beispiele zum Rechnen mit Logarithmen. Einleitend beschrieb er die spezifische Situation aus Sicht der Optik: Bereits früh habe man erkannt, dass die Berechnung von Optiken »schneller und billiger« erfolgte, als dies durch experimentelle Ermittlungen der Fall sein könne (Sonnefeld 1921, 126). Während der Physiker Ernst Abbe Anfang der 1880er die Mikroskop-Optiken noch selbst berechnete, übernahm der Physiker Siegfried Czapski ab 1885, als dessen Assistent, und der Mathematiker Paul Rudolph ab 1885 zunehmend auch Rechenaufgaben. Rudolph begann in den folgenden Jahren Rechner einzustellen, um eine Entlastung der Konstrukteure von den Rechenaufgaben zu erreichen. In den folgenden 30 Jahren stieg die Anzahl kontinuierlich auf 20 Rechner verschiedenster Qualifikationsstufen, verteilt auf die drei Abteilungen Geo/Teleskope, Astronomie/Mikroskopie/Messtechnik/Optotechnik und Fotografie (Fernr. V. 1928). In seinem Bericht gibt Sonnefeld einen subjektiven Einblick in den Berufsstand: »Ein zäher Wille, ein persönlicher Schliff, seine Gedanken streng auf eine Sache zusammenzufassen, ist beim Rechnen durchaus notwendig. Unruhige Geister, Phantasten, Träumer werden wohl kaum jemals rechnerisch etwas leisten. Versäumte Nachtruhe und selbst harmloseste Ausschweifungen machen sich bei keinem Beruf so stark bemerkbar wie bei den Rechnern« (Sonnefeld 1921, 126). Wie arbeiten die Rechner und Mathematiker der 1920er Jahre bei Carl Zeiss? Sonnefeld

---

<sup>149</sup> Die Arbeitsblätter sind nicht erhalten, sodass ihre genaue Beschaffenheit nur aus den heute zugänglichen Äußerungen de Pronys rekonstruiert werden konnten (Grattan-Guinness 2003, 108). Die von Grattan-Guinness verwandte Metapher des »Fließband« ist zur Zeit de Pronys schief, denn die Erfindung des Fließbandes in den Fabriken Henry Fords erfolgte erst Jahrzehnte später. Zu de Prony siehe auch *On the Economy of Machinery and Manufactures* (Babbage 1832, 153–158) und *Labor and Monopoly Capital – The Degradation of Work in the Twentieth Century* (Braverman 1974, 218–220).

<sup>150</sup> Zum Kontext: Eine für Deutschland prägende »Rechenfabrik« (Sobeslavsky 1996, 137) war das Institut für Angewandte Mathematik an der TU Darmstadt, welches ab 1928 aufgebaut wurde und zu angewandten Algorithmen forschte, aber auch über 70 Rechnerinnen verfügte, die angewandte Probleme der Mathematik für industrielle Auftraggeber rechneten (Biener 1999, 61). Den Zusammenhang zwischen Lochkartentechnik, Tabellen und angewandtem Rechnen als Fabrikarbeit in Deutschland stellt Richard Vahrenkamp in *Die erste Informationsexplosion – Die Rolle der Lochkartentechnik bei der Büro-rationalisierung in Deutschland 1910 bis 1939* (Vahrenkamp 2017, 216) her. Den Kontext des (west-)deutschen wissenschaftlichen Rechnens zwischen 1870 und 1970 beleuchtet *Rechner für die Wissenschaft – »Scientific Computing« und Informatik im deutschen Wissenschaftssystem 1870–1970* (Hashagen 2011).

vermerkt, obwohl sie zu dieser Zeit bereits in Massenproduktion sind, »teure Rechenmaschinen werden nicht gebraucht« (ebd.). Verwendung als Hilfsmittel für Multiplikation und Division finden Logarithmentabellen, die es erlauben, die Multiplikation in Additionen zu zerlegen, die Divisionen zu Subtraktionen. Diese Tabellen dienen als Werkzeug des Rechnens, organisiert als stabiler Speicher konstanter Werte. Auch die anderen verwendeten Tabellen haben Speichercharakter – eine besteht aus den Brechungsexponenten<sup>151</sup> verschiedener Glassorten, eine andere aus den Radien der Glasschalen, die in der Werkstatt vorhanden sind. Wie kommen diese zur Anwendung? »Im Gehirn des Rechners fest vermerkt sind dann noch eine große Zahl mathematischer Formeln. Das sind allgemeine Vorschriften in mathematischer Zeichensprache, nach denen die Rechnungen auszuführen sind« (Sonnefeld 1921, 128). Retrospektiv zwingt sich der Vergleich von Sonnefelds ›Gehirn des Rechners‹ zur Rede vom ›Elektronenhirn‹ auf, welche die 1950er durchzog. Laut Sonnefeld sind im Kopf des Rechners mathematische Formeln »fest vermerkt«, gespeichert. Von dieser Diktion ist die Idee eines Speichers, der an das Rechenwerk angebunden ist, nicht mehr weit. Denn auch dieser wird mit Formeln, bzw. Algorithmen und Daten gefüllt.



Abb. 26. Rechner bei Carl-Zeiss Jena: Gäste zur Jubiläumsfeier am 16. Oktober 1936 im Thüringer Hof anlässlich der 25-jährigen Betriebszugehörigkeit Dr. C. A. Sonnefelds (ZEISS Archiv, BACZ 10948).

<sup>151</sup> Die Lichtgeschwindigkeit im Glas im Verhältnis zur Lichtgeschwindigkeit in der Luft.

P.A. - Spiegel 260/62,5 mm <sup>360/62,5</sup> . Z. 269723.

H. H. 3. 132/4.  
Z 269723  
369724

Angeben für die Primärschablonen.

Φ	Tilbstrahlänge R	Paraxiallänge R	Φ	Tilbstrahlänge R	Paraxiallänge R	Φ	Tilbstrahlänge R	Paraxiallänge R
0°	560,00	556,60	11°	528,39	524,53	22°	468,80	464,74
0° 30'	559,93	556,53	11° 30'	525,87	521,98	22° 30'	466,16	462,11
1°	559,70	556,29	12°	523,31	519,39	23°	463,54	459,50
1° 30'	559,32	555,90	12° 30'	520,71	516,77	23° 30'	460,94	456,91
2°	558,80	555,37	13°	518,08	514,13	24°	458,37	454,34
2° 30'	558,13	554,69	13° 30'	515,41	511,45	24° 30'	455,83	451,80
3°	557,31	553,85	14°	512,71	508,74	25°	453,32	449,30
3° 30'	556,35	552,87	14° 30'	509,99	506,00	26°	448,37	444,36
4°	555,25	551,75	15°	507,25	503,25	27°	443,55	439,56
4° 30'	554,02	550,50	15° 30'	504,49	500,48	28°	438,86	434,89
5°	552,67	549,13	16°	501,72	497,70	29°	434,30	430,35
5° 30'	551,20	547,63	16° 30'	498,94	494,91	30°	429,86	425,93
6°	549,61	546,07	17°	496,16	492,12	31°	425,56	421,65
6° 30'	547,90	544,28	17° 30'	493,38	489,34	32°	421,39	417,50
7°	546,09	542,44	18°	490,61	486,56	33°	417,36	413,49
7° 30'	544,17	540,49	18° 30'	487,84	483,79	34°	413,46	409,61
8°	542,15	538,44	19°	485,08	481,02	35°	409,69	405,82
8° 30'	540,04	536,30	19° 30'	482,33	478,27	36°	406,05	402,18
9°	537,85	534,09	20°	479,59	475,53	37°	402,55	398,67
9° 30'	535,59	531,81	20° 30'	476,86	472,80	38°	399,18	395,36
10°	533,26	529,46	21°	474,15	470,09	39°	395,94	391,71
10° 30'	530,86	527,03	21° 30'	471,46	467,40	40°	392,82	388,42

für 260 mm Durchmesser + 40% sind die Werte bis einfl. 23° zu nehmen.  
 Um unempfindliche Reflektoren zu erhalten, ist die Anzeigefür die Paraxiallänge des 360 mm  
 Spiegel im Fernrohrverhältnis mit zwei Wechselböden mit bis 462,4 mm Durchmesser  
 versehen, was einer Fugbreite von 28,4% entspricht.

Jene, 13. 9. 28.  
 Bg. hof.

Abb. 27: Tabelle zur Berechnung der Primärschablonen von Parabolspiegeln (Sonnefeld 1928b, ZEISS Archiv, BACZ 19839).

Ein Übergang deutete sich an. Bereits 1928 schien an den Rechenmaschinen kein Weg vorbei zu führen, denn Sonnefeld berichtet in einem Schreiben an die Geschäftsführung: »Die Rechner Meoll, Kuhn, Schönberger, übrigens die besten Kräfte des Büros, haben den Vorteil des Rechnens mit der Maschine klar erkannt und würden nur mit Widerwillen zum Tafelrechnen zurückkehren« (Sonnefeld 1928a). Sonnefeld setzte die Leistungssteigerung gegenüber dem Rechnen mit Logarithmentafeln auf 30% an

und stellte eine Verminderung von Rechenfehlern fest. Gleichzeitig verschob sich die Art des Rechnens: Während vorher viele kleine Rechenschritte sukzessive im Kopf des Rechners abzarbeiten waren, zum Beispiel beim Rechnen mit Logarithmentafeln oder mit der Differenzmethode, ist nun ein Teil dieser Sukzession in die Maschine verlagert, sodass er »sich mehr auf Zweck und Gang der Rechnung konzentrieren« konnte (Sonnefeld 1928a). Die Arbeitsteilung verschiebt sich: jene algorithmischen Anteile, die bisher im Kopf, unter Zuhilfenahme von Papier reihenweise durchgerechnet wurden, werden abgegeben an die Maschine. Während die Bedienzeiten der Maschine zunahm, entfiel die Zeit für das Nachschlagen von Logarithmen in den tabellarischen Tafelwerken. Eines jedoch blieb gleich: Da ein Großteil der Ergebnisse zur Verwendung in den Werkstätten und Entwicklungsbüros dienen sollte, wurde er in tabellarischer Form auf Papier niedergelegt (Abb. 27).

Immo Kerner, Mathematiker bei Zeiss, beschrieb Ende der 1950er Jahre, in einem Vortrag vor Mathematik-Studenten und Lehrenden der Universität Leipzig, die Unterscheidung zwischen Mathematiker und Rechner: »Es ist nämlich für die Praxis ein Problem durchaus nicht abgeschlossen, wenn das Ergebnis in Form einer oder mehrerer Gleichungen implizit oder explizit vorliegt, sondern es muß fertigungsgerechnet und zahlenmäßig geliefert werden. Oft sind hierzu umfangreiche Zahlenrechnungen nötig, kein Mensch erwartet vom Mathematiker, daß er diese Rechnungen selbst ausführt, aber er muß dazu in der Lage sein. Nur dann kann er geeignete Rechenvorschriften für den technischen Rechner entwerfen und nur dann kann er auf der Tischrechenmaschine Proberechnungen durchführen, ohne ein mitleidiges Lächeln zu erregen« (Kerner 1959, 28). Kerner deutete hier eine Abstufung der Arbeitsteilung an, die bei Carl-Zeiss Jena auch in einem entsprechendem Dokument festgehalten ist.<sup>152</sup>

Während die Rechner einzelne Berechnungen laut Anweisung vornehmen, ist die zweite Gruppe der Mathematiker für die Problemanalyse und Aufstellung der Rechenanweisungen zuständig. Die arbeitsteilige Trennung in niedriger qualifizierte Rechner und höherqualifizierte Mathematiker ähnelt vielen Rechenbüros, und dies nicht zufällig: In der einschlägigen Fachliteratur wurde diese arbeitsteilige Praxis angeraten (vgl. Grier 2005). Die akademischen Mathematiker bei Carl-Zeiss benötigten neben tiefergehenden mathematischen Kenntnissen auch Wissen jenseits ihres Fachgebiets, denn sie arbeiteten interdisziplinär mit Physikern, Chemikern und anderen Wissenschaftsfeldern zusammen. Dies schien in der akademischen Ausbildung der angewandten Mathematiker der 1950er in Leipzig ein Manko zu sein. Immo Kerner wies in seinem Vortrag vor Alumni, Studierenden und Lehrenden des Studienganges mit Dringlichkeit darauf hin, dass hier nachgebessert werden müsse.

Kerner befand sich mit seiner Darstellung bereits am Übergang von den Rechentätigkeiten mit Hilfe mechanischer Rechenmaschinen zu elektronischen Rechenmaschinen, wie jener Optik-Rechen-Maschine Oprema, die seit 1955 bei Carl Zeiss in Betrieb war und die erste elektro-mechanische Universalrechenmaschine der DDR darstellte.<sup>153</sup> Er konstatierte eine deutliche Zunahme von »Massenrechnungen« nach dem Ende des 2. Weltkrieges, vor allem in Verwaltung und Statistik. »[...] die verwaltungsmäßigen Aufgaben schwollen derart an, daß man auf Grund des Zahlenmaterials und mit Hilfe der vorhandenen Rechenmittel viel zu spät den wahren Stand der Produktion erfuhr, so daß ein operatives Eingreifen nicht mehr möglich war« (Kerner 1959, 30).

---

<sup>152</sup> Darin angegeben: Technischer Hilfsrechner, Technischer Teilrechner, Technischer Vollrechner und die wissenschaftliche Mitarbeiter der Stufen Technischer Assistent, Wissenschaftlicher Assistent und Wissenschaftlicher Abteilungsleiter (EboMath/Hass/Kul 1953).

<sup>153</sup> Zum Vergleich: Max-Planck-Institut Göttingen G 1 (1952), Zuse Z5 für Leitz, Wetzlar (1953), Oprema für Carl Zeiss, Jena (1955), Göttingen G 2 (1955), Darmstädter Elektronischer Rechenautomat für TU Darmstadt (1957), Technische Hochschule München PERM (1956), Technische Hochschule und VEB Robotron Dresden D1 (1956).



Im wissenschaftlichen Bereich nahmen die Aufgaben derart an Komplexität zu, dass sie zwar überschaubar aber von menschlicher Hand nur unter großem Zeitaufwand lösbar waren. Zu den 1951 bearbeiteten Problemen zählten beispielsweise die Elastizität des Federkippspiegel oder die Fehleruntersuchung an einem Zweipunkt-Passimeter, also nicht allein optische, sondern auch komplexe feinmechanische Aufgaben (EboMath 1951).

Noch eine weitere Quelle gibt Einblick in den Übergang vom Kopf- zum Maschinenrechnen. Der Ingenieur und Mathematiker Wilhelm Kämmerer, der die Oprema entwickelte, gab 1956 einen *Ausführlichen Abschlußbericht zur Forschungs- und Entwicklungsarbeit* an die Geschäftsleitung des VEB Carl-Zeiss Jena. Zwei Rechner des Entwicklungsbüros Technik (EBoTh) wurden beauftragt, gegen die Oprema anzutreten. Als Aufgabe wählte er die Berechnung eines Polynoms fünften Grades,<sup>154</sup> das als Reihe von 151 Werten in einer Tabelle dargestellt werden sollte. Dabei kam den Rechnern eine mechanische Mercedes-Euklid Rechenmaschine zur Hilfe. »Die Koeffizienten mußten mittels einfacher algebraischer Beziehungen ermittelt werden. Die Abszissen selbst sollten als Glieder einer alternierenden geometrischen Reihe berechnet werden. Diese Form wurde gewählt, um dem Rechner nicht den Vorteil einer Routinearbeit zu geben, sondern gerade Einblick in die tatsächliche Leistungsfähigkeit im Rahmen eines 8½ Stundentages zu gewinnen« (Kämmerer 1956, 6).

Folgt man dem Bericht<sup>155</sup>, so sprachen die Ergebnisse eine deutliche Sprache: »Beachtenswert ist die Tatsache, daß im Mittel 10% aller Werte fehlerhaft blieben, obwohl der Gang der Werte eine gute Kontrollmöglichkeit bot. Erst der Ansatz zweier Rechner gab dem Büro die Möglichkeit, die Fehler zu erkennen und zu beseitigen« (Kämmerer 1956, 7).

Wenn wir uns die maschinische Struktur der Oprema ansehen, finden wir eine ganze Reihe weitere Kontinuitäten entlang der Organisation des Rechengangs menschlicher Rechner. Die Oprema war strukturell aufgebaut aus

- einem addierenden Rechenwerk – dem prozessierenden Gehirn,
- einem zyklischen Speicher, der rotierend verschiedene Werte zur Verfügung stellen kann und damit Reihungen erlaubt und einem konstanten Speicher, in dem wir die ehemals tabellarisch gespeicherten Brechungskoeffizienten vermuten können, welche über Speicher-Adressen ansteuerbar waren,
- einem Steuerwerk, das die Abfolge der Rechenschritte steuert – der Algorithmus, beziehungsweise die »Rechenfolge«, die zuvor im Kopf zu behalten war,
- dem Register – als Zwischenspeicher beziehungsweise Kurzzeitgedächtnis,
- sowie der Ausgabe auf einem Drucker, welcher per Festschriftschrift (engl. Monospace) ein entsprechendes Raster über die Seite legte

---

<sup>154</sup> Ein Polynom besteht aus eine Reihe von Variablen, deren Potenz gezogen wird und die miteinander addiert werden. Bei einem Polynom fünfter Ordnung ist die höchste Potenz 5, z.B.  $P(x) = 2 \cdot x^5 - 7 \cdot x^4 + x^3 + 2 \cdot x^2 + x + 8$ . Als Rechenverfahren kam das sogenannte Horner-Schema zur Anwendung, wobei die Potenzen in die einfacher zu berechnenden Multiplikationen und Summen zerlegt werden, zum Beispiel in:  $x \cdot (x \cdot (x \cdot (x \cdot (2 \cdot x - 7) + 1) + 2) + 1) + 8$ . Um im Kopf zu berechnen, wird für gewöhnlich auf eine tabellarische Darstellung zurück gegriffen, das sogenannte Horner-Schema.

<sup>155</sup> Wenn auch im Bericht vereinzelt auf Probleme der Arbeit mit der Oprema eingegangen wird, erscheint er insgesamt äußerst optimistisch. Aus anderen Berichten von Rechenmaschinen dieser Zeit ist die ständige Fehleranfälligkeit im Betrieb bekannt und der Aufwand für die Erstellung der Algorithmen für die Maschine und das Eingeben, das Programmieren, nahm spürbaren Raum ein. Zumindest deutet Kämmerer an, wie im Testbetrieb eine der Maschine angepasste Algorithmisierung zunehmend verfeinert wurde. »Daneben begannen auch Erprobungen eines ersten Optik-Programms, das zunächst noch eine Reihe von Schwächen zeigte, so daß die Einsatzmöglichkeiten der Oprema nicht voll ausgenutzt wurden. Das Programm wurde stärker konzentriert« (Kämmerer 1956, 8).

Diese Analogien (vor allem Rechenwerk–Gehirn) sollen nicht anthropomorphisierend gelesen werden, denn das Anliegen dieser Studie liegt nicht in der Vermenschlichung der Maschine, sondern in der Fragestellung, wie menschliche Arbeit, in diesem Falle Kopfarbeit, maschinisiert wurde und welche Rolle dabei die tabellarische Erfassung als formatierende Informationsstruktur spielte.

Kämmerer rechnete in seinem Bericht die Arbeitsstunden der menschlichen Rechner auf die Rechenmaschine um und kam zu dem Schluss, das für 24.000 Rechenstunden, die innerhalb von 10 Wochen ansetzbar wären, äquivalent 83 Rechner benötigt würden. »Es kann als sicher angenommen werden, daß die Opremaarbeiten 1/3 so teuer zu stehen kommen, als wenn die gleiche Arbeit von Optikrechenern erledigt würde. [...] Wie stark der Bedarf einer solchen Rechenanlage im VEB Carl Zeiss Jena ist, ergibt sich aus der Tatsache, daß für die Planung 1956 die Kapazität der Oprema fast restlos durch innerbetriebliche Optikaufgaben in Anspruch genommen ist« (Kämmerer 1956, 10).

Diese Art der Rechenleistungsschätzung ist nicht nur für die Oprema zu beobachten, sie ist beispielsweise auch für das New Yorker Mathematical Tables Project der 1930er–1940er Jahre belegt. Der angewandte Mathematiker L. J. Comrie, ein starker Vertreter des Einsatzes von Rechenmaschinen, bezeichnete das Mathematical Tables Project als »das mächtigste Rechenteam, welches jemals existierte«,<sup>lxxxix</sup> und John von Neumann verfolgte einen Teil der Arbeiten, um abzuschätzen, wieviel Rechenzeit der ENIAC-Computer für ein spezifisches Problem benötigen würde (Swade 2003, 288).

Zusammenfassend tritt De Pronys Geschichte der Tabellen als eine der Umbrüche, des Wandels und der Misere auf: Die Arbeit, die ihm 1790 »aufgebürdet« wurde, drohte ihn von anderen Unternehmungen abzuhalten; im post-revolutionären Frankreich ist die staatliche Ordnung eingeschränkt und hindert seine Untergebenen an ihrer Vermessungstätigkeit; seine Position in Paris ist teilweise prekär, eigentlich soll er in der Provinz eingesetzt werden. Diese Misere jedoch ist Ausgangspunkt einer methodischen Innovation, in der die Tabelle als Werkzeug und Speicher und die Arbeitsteilung als Prozess wesentliche Rollen spielen.<sup>156</sup> Dabei wird die Kopfarbeit delegiert an das Planungsteam und von diesem so aufbereitet, dass sie nun mehr als reiner Automatismus, beziehungsweise als Algorithmus in den Köpfen und Händen der Rechner existiert.<sup>157</sup> Diese Innovation ist situiert am Übergang vom Ancien-Regime zur bürgerlichen Republik, der die Volkswirtschaft dahingehend verändert, als dass diese nicht mehr zentral auf die Luxusproduktion für den König ausgerichtet ist und ein von dessen Interessen unabhängiger »freier« Markt entsteht.

Für die Arbeit der menschlichen Rechner (nicht der Mathematiker) bei Carl Zeiss Jena ist es retrospektiv möglich, die mechanisierte Kopfarbeit der Computer als organisatorische Inspiration für mechanische und elektronische Rechenmaschinen zu interpretieren (vgl. auch Swade 2003). Ziel war die maximale Rechenqualität bei höchster Rechenquantität. In dem Moment, indem die von den Arbeitern verausgabte Arbeit zu teuer wurde, konnten sie durch Maschinen ersetzt werden. Die signifikante Wirkmächtigkeit der Methode Arbeitsteilung unterstreicht der Sozialwissenschaftler und Ökonom Herbert Simon, welcher nicht ohne Söffizanz zur Genese elektronischer Rechenmaschinen anmerkte, dass »Physiker und Ingenieure wenig mit der Erfindung des digitalen Rechners zu tun hatten – dass der wirkliche Erfinder der Ökonom Adam Smith war« (Simon/Newell 1958, 2).<sup>xc</sup>

---

<sup>156</sup> Markus Krajewski schildert eine ähnliche Situation, die fast komische Züge annimmt – die Erfindung des losen Karteikastenkastens, in Abkehr vom gebundenen Katalog durch William Crosswell, aus einer Situation heraus, in der er nach langen Phasen des Nichts-Tuns – »February 23, 1813 At the Lib – do nothing.« – schnellstmöglich Ergebnisse in Form eines Bibliothekskatalog liefern sollte (Krajewski 2011, 70–78).

<sup>157</sup> Man kann argumentieren, dass dies der Zweiteilung in 1.) Programmierer und 2.) Rechenmaschine ab den 1930ern entspricht und die menschlichen Rechner als Mechanismus oder Maschine interpretieren.

	<b>Rechner</b>	<b>Planer</b>
<b>Arbeitsform</b>	Fabrikarbeit / Mechanisierte Kopfarbeit	Wissensarbeit
<b>Arbeitskräfte</b>	Dutzende bis Hunderte	wenige
<b>Qualifikation</b>	niedrig	hoch
<b>Tätigkeitsfeld</b>	extrem reduziert	vielfältig
<b>Maschinisierung</b>	möglich	kaum; später entwickelte sich das Berufsbild des Programmierers

Tabelle 3: Arbeitsteilung in de Prony's und den Carl-Zeiss »Rechenfabriken« (Autor).

Nimmt man diese politökonomische Wendung zur Kenntnis, macht es Sinn, nach den Registern der angewandten Mathematik im Sinne der geteilten Arbeit zu fragen. Dabei werden Kontinuitäten und Diskontinuitäten zwischen menschlichen und maschinellen Computern sichtbar, die sich am jeweiligen Arbeitsbereich scheiden. Während das Rechnen im Kopf mechanisiert werden konnte, ist die Aufgabe des Planens, der Algorithmisierung im Berufsbild der Informatik aufgegangen. Verbleiben wir noch kurz im politökonomischen Register: In der *Encyclopedia Britannica* von 1911 wurden Produktivitätskriterien mathematischer Tabellen angeführt, die aus ihrem Charakter als Wissensspeicher folgen. So argumentiert der Verfasser des Eintrages, der Mathematiker und Astronom, James W. Glaisher »Der Wert einer Tabelle ist nicht von der Problematik ihrer Berechnung abhängig; da sie, sobald sie einmal hergestellt wurde, dauerhaft existiert und der Arbeitsaufwand zu ihrer Erstellung, zumindest aus Sicht der Nutzer, unerheblich ist« (Glaisher 1911, 325).<sup>xc1</sup> Einmal erstellte mathematische Tabellen ersparten eine erneute Berechnung und die damit verbundene Arbeit, sie verbilligten und beschleunigten die Produktion jenes Wissens, das auf vorhandenen Tabellen aufbauen konnte.<sup>158</sup> An diesem Beispiel sehen wir bereits die Diskussion um den Wert des Wissens heraufziehen. Dieser wird in der *Encyclopedia* so angegeben, dass das Wissen nach einmaliger Produktion vielfach verfügbar und anwendbar ist. Daraus ergäbe sich, dass mit jedem Gebrauch eines Tabellenwerks sich dessen Anschaffungskosten aus Nutzersicht verringern. Jede unveränderte Auflage eines Tabellenwerks würde hingegen aus Produzentensicht die Produktionskosten minimieren.

Die mathematische Tabelle stand somit im Spannungsfeld zwischen einerseits dem mathematischen Werkzeugcharakter in der Herstellungsphase mathematischer Berechnungen und andererseits der Funktion als Speichermedium mathematischer Ergebnisse. Ihre arbeitsteilige Herstellung beeinflusste maßgeblich Ideen zur Strukturierung früher elektronischer Rechner (Tabelle 3).

#### 2.2.2.5 Zwischenfazit

Diese vier Tabellenarten – zur Wissensordnung; für Statistik, Organisation und Prozesse; zur Transaktion und Kooperation und als Mathematische Tabellen – sind bei weiterer Forschung vermutlich um weitere Kategorien zu erweitern und auszudifferenzieren. Es konnte verdeutlicht werden, wie das tabulare Operieren die Tiefengeschichte der Informationsverarbeitung geprägt hat.

<sup>158</sup> Gemeint sind damit die gedruckten Tabellenwerke, zum Beispiel die Logarithmen-Tabellen. Die Gegenbewegung dazu markiert der Umstand, dass bei Carl-Zeiss die Optik-Berechnungen für jede Glas-Charge abhängig von deren Qualität neu durchzuführen war.

```

CREATE TABLE students
(
    name                TEXT,
    studierenden_id     INTEGER    PRIMARY KEY,
    geburtsdatum        DATE,
    benotung            INTEGER    NOT NULL,
    bestanden           BOOLEAN    NOT NULL
);

```

Abb. 28: Beispiel für das Erstellen einer Tabelle in SQL mit Hilfe des Befehls ›Create Table‹. Die Studierenden-ID wird als Primärschlüssel festgelegt und ist eine mehrstellige Zahl, die Spalte Geburtsdatum wird im Datumsformat angelegt, die Benotung als Zahl und Bestanden als Wahrheitswert „wahr/falsch“. Benotung und Bestanden dürfen zudem nicht leer, NOT NULL, bleiben und müssen bei Eingaben einen Wert erhalten (Autor).

Als Kulturtechnik vereint das Tabellieren verschiedene menschliche Fähigkeiten: einerseits maßgeblich das Verfügen über Schrift, Lesen und Schreiben und andererseits über die räumliche Organisation von Gegenständen in der tabellarischen Fläche, welche dem Umgang mit Bildern entstammt. Sie stellt eine gleichermaßen sinnliche als auch abstrahierende menschliche Tätigkeit dar. Das hybride Verfahren der Tabulation, eine wissensproduzierende Technik aus Bildlichkeit und Schriftlichkeit, so ist zu konstatieren, ist medientheoretisch bisher nur am Rande zur Kenntnis genommen worden. Dem ist seine zentrale Bedeutung für Wissensorganisation, Wissensprozession und die Erzeugung neuen Wissens entgegenzusetzen und diese zentrale Bedeutung setzt sich in der Folge in die Tiefen der Datenbankmanagementsysteme und vor allem des relationalen Modells fort (Abb. 28). Deutlicher herausgearbeitet werden soll dieser Zusammenhang im nächsten Abschnitt.

### 2.2.3. Am Übergang zu Datenbankpraktiken

»Als Sharon Weinberg, Ted Codd und ich unsere Beratungsfirma gründeten, kam die Frage eines Firmenlogos für die Verwendung auf Visitenkarten und ähnlichem auf. Ich sagte, dass meiner Meinung nach, das Logo das stilisierte Bild einer Tafel beinhalten sollte. Sharon stimmte zu. Ted sagte: ›Verstehe ich nicht – wir sind doch nicht im Möbelgeschäft, oder?‹« (Date 2006, 10).<sup>xcii</sup>

»Das Ziel war es, den Nutzern ein konzeptuelles Modell anzubieten, welches ohne Überraschungen war – man nannte es das Prinzip der geringsten Überraschung. Unser [Visicalcs] Modell war die Tabellenkalkulation – eine einfaches Papierraster welches auf dem Tisch ausgebreitet wird« (Frankston 2015).<sup>xciii</sup>

Ausgangspunkt der Überlegungen war die Frage nach der historischen und kulturtechnischen Herkunft der Tabelle und nach der Operationalität des Tabellierens. Im Folgenden soll in mehreren Schritten ihre Bedeutung als Grundlage elektronischer Datenbanken angezeigt werden, indem Prozessualität, Operationalität, Diskursivität für die tabellarische Relation im Übergang zur Datenbankfiguration neu

gefasst werden. Sie kennzeichnen, auf welche Weise die Tabelle grundlegende Materialität relationaler Datenbankmanagementsysteme werden konnte.<sup>159</sup>

### 2.2.3.1 Prozessualität und Leere

Ein erster, bereits in den Tabellenpraktiken angesprochener, Punkt betrifft die Frage der Leere. Aristoteles' Überlegungen eines Äthers, auf die Leere des physikalischen Raumes bezogen, wurden im Mittelalter als Angst vor der Leere gekennzeichnet. Leer, das ist der Ort an dem etwas sein könnte, aber nichts ist, so beschrieb es Aristoteles und das könne nicht sein.<sup>160</sup> Daraus schlossen seine frühmittelalterlichen Wiederentdecker Thomas von Aquin und dessen Zeitgenossen, dass die »Natur das Nichts verabscheut«. Dieser Horror Vacui verliert sich in der europäischen Denktradition erst im 17. Jahrhundert mit den Vakuum-Experimenten, von denen Otto von Guericke das populärste durchführte. Wenn man erneut in Aristoteles Diktion hineinhört – sofern das mit der zeitlichen Distanz überhaupt möglich ist – so spricht er auch von einem Potenzial, vom Ort, an dem etwas sein könnte oder sein sollte. Das mediale Verfahren der Leere ist bekannt: Kasimir Malevič's Schwarzes Quadrat auf weißem Grund, die monochrome-blauen Bilder und der leere Raum im Museum Haus Lange in Krefeld von Yves' Klein (1961) erinnern uns an die Leere des Raumes, die sich bei genauerer Betrachtung weniger leer darstellt, als ursprünglich angenommen. Klein hatte den ›leeren‹ Raum, mit dem Farb-Roller für die Ausstellung geweißt und so zum Gegenstand gemacht. Malevič hat nach dem schwarzen (1915), ein rotes Quadrat (1915) und ein weißes Quadrat auf weißem Hintergrund (1919) gemalt.

Jedem sind durch den Maler selbst, durch Kunsthistoriker und durch nachfolgende Künstler Bedeutungen zugeordnet worden (Arns 2004). Als musikalisches Werk ist John Cages 4'33 Stille (1952) berühmt geworden. In der Schrift begegnen wir mit jedem Leerzeichen, welches die Worte voneinander abgrenzt, einer Leere der Grammatik. »Der Unterschied zu Bildern liegt auf der Hand: Schriften bedürfen der Leere und der Lücke; würden gedruckte Buchstaben so weit zusammengedrückt, dass schließlich eine einzige schwarze Fläche entsteht, so haben wir zwar ein monochromes Bild gewonnen, aber dafür die Schrift und den Text verloren« (Krämer/Brandner 2013).

Die Leere jedoch, die der Tabellenzelle am nächsten kommt, ist jenes Planquadrat, welches in frühen Kartografien leer blieb, weil es den Vermessern unbekannt war.<sup>161</sup> Darauf hatte bereits Lewis Carolls Illustrator Henry Holiday in *The Hunting of the Snark* durch eine entsprechende Abbildung hingewiesen, die ein kartographisches System und Himmelsrichtungen anzeigte, sonst jedoch leer blieb (Siehe Abb. 29). Mit zunehmender Vermessung der Welt zeigte das leere Planquadrat den Ozean an, der scheinbar nichts Kartographierenswertes enthält.

---

<sup>159</sup> Zwei weitere grundlegende Materialien des relationalen Modells verfolge ich an anderer Stelle: Die algorithmische Ausformulierung der relational gewendeten Mengentheorie und die operationale Logik der Computersprache, konkret von SQL, diskutiere ich im Abschnitt 1.2.5. *Die relationale Sicht auf Informationen und normalisierte Beziehungen*.

<sup>160</sup> Siehe *Physik. Vorlesung über die Natur* (Aristotle/Zekl 1987, 175–197).

<sup>161</sup> Das leere Feld einer Tabelle in der Zeile oder in der Spalte könnte im weiteren Sinne auch als Zwischenraum aufgerufen werden, für welchen Dotzler/Schmidgen konstatieren: »Zwischenräume sind jene Schnittstellen, Intervalle und Abstände, in denen sich elementare Prozesse der Wissensproduktion ansiedeln« (Dotzler/Schmidgen (Hrsg.) 2008, 7).

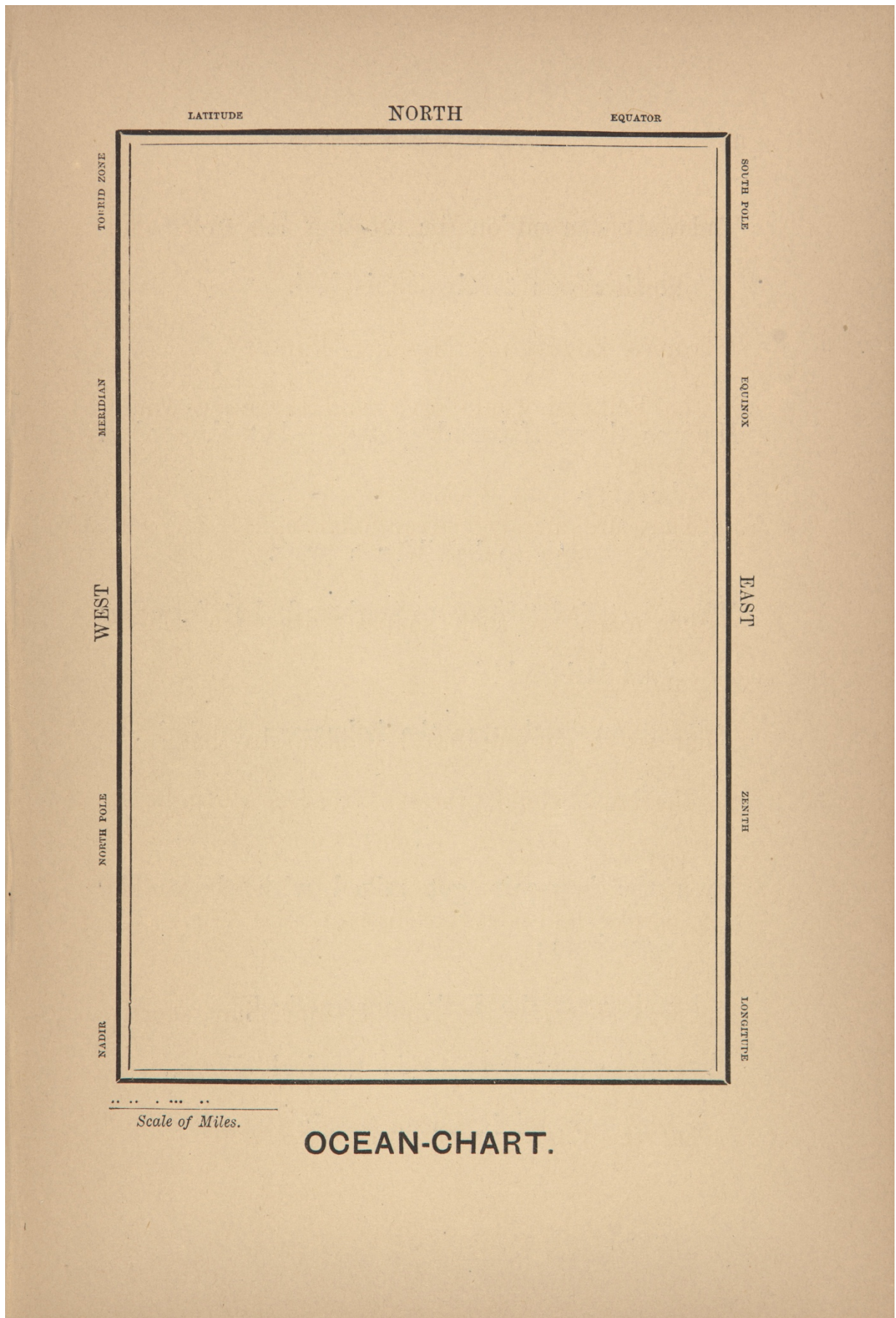


Abb. 29: Konstruktion der Leere: »eine Karte, welche die Schiffsbesatzung verstand« (Carroll 1874).

Ebenso wie die kolonialen ›Entdecker‹ ihr Handlungen darauf abstimmen, die unerträgliche Leere zu füllen – zur Not mit den imaginierten Mountains of Kong, die im 19. Jahrhundert Karten des westlichen Afrika mit einem Phantasma füllten – erzeugt die leere Tabellenzelle die Notwendigkeit der Daten-Erfassung. Auf eines jedoch macht das satirische Beispiel aus *The Hunting of the Snark* umso deutlicher aufmerksam: Es bedarf eines konstruktiven Rahmens, um Leere anzuzeigen. In der Tabelle ist dies die Matrix der Lineatur.<sup>162</sup>

ID	post_author	post_date_gmt	guid	post_content	post_title
260	1	2019-08-17 19:38:09	http://www.irmielin.org/index.php/2019/08/17/	Making sure not to forget a single paper slip, Croswell glues every scrap of...	Dokumentation Intern

Abb. 30: Auszug aus der Wordpress-Tabelle wp\_posts. Die ID bezeichnet den Schlüssel, bzw. die Adresse des Artikels, in diesem Falle ›260‹. Unter post\_author ist die ID des Autors eingetragen, dessen Details jedoch in einer anderen Tabelle namens wp\_users zu finden sind. In der GUID ist die öffentlich sichtbare Publikationsadresse, die URL vermerkt und die Spalte post\_content enthält den kompletten Inhalt, während post\_title den Titel speichert (Screenshot aus Sequel Pro, Autor, 22.8.2019).

Ein praktisches Beispiel soll illustrieren, wie Leere ein Erfordernis von Formularen und Tabellen in Bezug auf Datenbanken ist: Wordpress, eine bekannte Content-Management Software für Websites besteht

- aus einem für die Leser sichtbaren ›Frontend‹ mit den Inhalten,
- aus einem für die redaktionellen Nutzer zugänglichen ›Backend‹, welches sich durch eine Reihe von zu füllenden Formularen auszeichnet, zum Beispiel für Überschrift und Inhalt, aber auch für Meta-Daten wie Publikationsdatum, Zugehörigkeit zu anderen Einträgen et cetera und
- aus einer ›Datenbank‹, die zwischen Frontend und Backend vermittelt.

Die gefüllten, (oder teils weiterhin leeren) Zellen einer Zeile in den Datenbank-Tabellen (Abb. 30) werden beim Aufruf der Wordpress-Website durch das Frontend aus der Datenbank ›gezogen‹, mithilfe von HTML und CSS formatiert und via Browsersoftware in eine spezifische, visuelle Form gebracht, die in ihrer Gesamtheit als Website wahrgenommen werden kann. Die Tabellen sind durch Spalten vorstrukturiert und mit jedem Anlegen eines neuen Artikels erhalten sie einen neuen Zeileneintrag, der Spalte für Spalte gefüllt wird und zwar einerseits durch Algorithmen (die beispielsweise automatisiert das Erstellungsdatum einfüllen) und andererseits durch Nutzer, welche durch die Formulare des Backends die Inhalte einpflegen und mit Hilfe des Speichern-Buttons in der Datenbank ablegen. Im Tabellenraum des relationalen Datenmodells, der tabellarischen Relation, trifft das Konzept von Leere unmittelbar auf Verkomplizierungen: So ist unter den Pragmatikern der SQL-Implementierung umstritten, ob leere Zellen leer bleiben sollten, oder besser als solche gekennzeichnet werden sollen, z.B. durch die Zeichenkette NULL.<sup>163</sup> Dies führt zu dem Paradox, dass eine beschriebene Zelle als leer anzusehen ist. Doch die NULL-Zelle repräsentiert ein Etwas: Zu verdanken ist diese Erkenntnis George Boole, dessen Mengentheorie, für Aristoteles unvorstellbar, die Möglichkeit einer leeren Menge

<sup>162</sup> Gitelman merkt an, dass Leere markiert werden muss, zum Beispiel indem leere Buchseiten mittels Druckerschwärze als solche markiert werden (Gitelman 2014, 23). Steiner vergleicht das Tabellengitter exemplarisch mit dem Koordinatengitter, das am Beginn der modernen Kartenproduktion steht, und es erlaubte, die bekannte Welt nach den Prinzipien der euklidischen Geometrie zu rationalisieren und vor allem jene leeren Bereiche des Gitters sichtbar machte, die es als »unentdeckt« zukünftig zu beschreiben galt (Steiner/Benjamin 2008, 284).

<sup>163</sup> Die Verwendung von NULL in SQL wurde durch Codd kritisiert, und diese Kritik erneuert durch Hugh Darwen.

vorsieht und damit logischer Operationalisierung unterwirft. Die Verwendung von NULL zielt darauf ab, eine Zelle als bearbeitet zu markieren: »Hier wurde ein Wert eingetragen!«<sup>164</sup> In gebräuchlichen Tabellen auf Papier hätte man diese NULL-Zellen durch eine diagonale Linie gesperrt.

Das Tabellieren ist auch ein Verfahren, um durch das Hinzufügen neuer Zeilen und neuer Spalten, Leere zu erzeugen. Die Leere thematisiert das Nicht-Gewusste, das Nicht-Erledigte und das Nicht-Berechnete und gibt Anlass für neue Vorgänge der Registratur und Identifikation, der Delegation und der Kooperation. In Tabelle und Datenbank ist die Leere nicht leer.

### 2.2.3.2 Die Tabelle als Diskurs

Es bleibt die Frage zu klären, warum und inwiefern Tabellen (und mit ihr Datenbanken) in den Diskursen wirkmächtig werden. Die Tabelle ist ein Spezifikationsraster, das die Gegenstände des Diskurses neu ordnet. Die daraus resultierende Darstellung ist nicht-identisch mit dem ursprünglichen diskursiven Feld, dem die Gegenstände der Darstellung entnommen wurden. Genau in dieser Ablösung von der Realität liegt der Werkzeugcharakter begründet. Das Tabellieren erzeugt eine Distanz zum Repräsentierten: die Repräsentation kann Operationen unterzogen werden und die distanzierten Informationsobjekte neu geordnet werden. Dies ermöglicht Szenarien möglicher Zustände, die im Tabellenraum simuliert werden, als Ausgangspunkt um Alltagshandlungen entlang der Zielvorgaben anzupassen. In diesem Sinne ist das Tabellieren eine Praxis zur gezielten Manipulation/Simulation des Verhältnisses von Signifikat zu Signifikant. Jede spezifische Konfiguration einer Tabelle oder Datenbank, insbesondere der Beschriftungen, der Untergruppierungen, die Entscheidung überhaupt darüber, wie Spalten oder Zeilen benannt werden sollen, um deren Informationsgehalt zu signalisieren, ist Diskurs. Tabellen/Datenbanken bezeugen eine Entwicklung, in der Teile von Realität nach der Maßgabe erhoben wurde, epistemologisch als Datum prozessierbar zu sein. Entlang der kartesischen Ratio etablierte sich dabei die Abspaltung jener Anteile von Realität, die dem Prozessieren abträglich sind. Ihre rigide Matrix erlaubt es, so disparate Felder wie politische Ökonomie, Policywissenschaften, Staatswissen und eine politische Enzyklopädie zu einander in Relation zu setzen, sodass diese überhaupt als Regierungswissen verfügbar werden.<sup>165</sup>

Das Tabellieren als Diskurs wurde hier, bezogen auf verschiedene Zeiträume, denen die soziale Form der Disziplinargesellschaft gemeinsam waren, untersucht. In der Theoretisierung der Kontrollgesellschaft von Gilles Deleuze spielt die Tabelle/Datenbank eine sich verändernde Rolle. Verorten lässt diese sich in der ausgeweiteten Verfügbarmachung und Neukombination von Informationsobjekten durch maschinelle Verarbeitung und der Flexibilisierung ihrer Speicherung. Mit dem knappen Satz »Die Individuen sind zu »Dividuen« geworden« (Deleuze 1992, 5),<sup>xciiv</sup> spricht Deleuze den Subjekten einen neuen Status zu, der aus Perspektive der relationalen Datenbank mit Poster (1995), insbesondere dem Kapitel *Databases as Discourse, or Electronic Interpellations*, sowie Raunig (2015) und Ott (2015) fortführend zu diskutieren ist. Deleuzes' Satz findet allerdings seine Fortsetzung: »und die Massen zu Daten, Märkten oder Banken« (Deleuze 1992, 5).<sup>xcv</sup> Auch wenn man in dieser Ergänzung bei »Banken«

---

<sup>164</sup> Ein mit »0« beschriebene Feld signalisiert mehr, als nur den Stand der Bearbeitung. Die Null ist in eine Zahl, welche arithmetische Operationen ermöglicht. Ein Beispiel, um den Unterschied zu markieren: Der Mittelwert berechnet sich aus der Summe der Elemente geteilt durch deren Anzahl. Ein mit »0« beschriebenes Feld erhöht die Anzahl der Elemente, während ein leeres Feld im Sinne von » « oder NULL in der Berechnung des Mittelwertes arithmetisch unbeachtet bleibt.

<sup>165</sup> Siehe auch *Kalkül und Leidenschaft – Poetik des ökonomischen Menschen* (Vogl 2002, 241–246).



zuerst an Finanzinstitute denkt, liegt die eigentliche Bedeutung als »Datenbanken« nahe. Das durch Deleuze angesprochene Gewordensein gilt es, weiterer Befragung zu unterziehen.<sup>166</sup>

Bei allem berechtigten Interesse am Theorem der Kontrollgesellschaft, wäre es ignorant zu behaupten, diese habe die Disziplinierung durch Institutionen oder die Selbstdisziplinierung der Individuen abgelöst. Die Verwendung der Tabelle in relationalen Datenbankmanagementsystemen zeigt beispielhaft Kontinuitäten an: sie koppelt die Disziplinargesellschaft an die Kontrollgesellschaft.

### 2.2.3.3 Zwischen Abfrage und Menge: Oberflächen und Tiefenschichten

Daten, Informationsmodelle und Algorithmen, die in dieser Studie als Momente von Schriftlichkeit verstanden werden, sind wesentliches Moment der Tiefenstruktur elektronischer Rechenmedien. An die bildliche-schriftliche Oberfläche heutiger Graphical User Interfaces können sie in vielfacher Form treten. Für den ›Maschinenraum‹ konstatierte Marcus Burkhardt im Verfolgen der Frage nach dem Erfolg des relationalen Datenbankmodells: »[Es verfügt] mit der Tabelle als *conceptual representation* der Tiefenstruktur der Datenbank über eine geeignete Denkfigur, welche Nutzern das Nachdenken über Datenbankoperationen und das Arbeiten mit Datenbanken weitgehend ohne mathematisches Vorwissen ermöglicht« (Burkhardt 2015).

Genau jenes Register, »ohne mathematisches Vorwissen«, war für den Mathematiker Edgar F. Codd anfangs ein blinder Fleck. Die maßgebliche Veröffentlichung zum relationalen Datenbankmodell 1970 argumentierte allein mit der mathematischen Logik der Mengentheorie. Die Tabelle trat zwar implizit in Abbildungen und als Denkfigur auf, aber ihren Wert als Interface erkannte Codd noch nicht (Abb. 31). Seine Abhandlung wurde teils als »obskur«, mit »komplizierten« mathematischen Formeln, wahrgenommen. Die Erkenntnis der Metaphernkraft des tabellarischen Raums ist vielmehr Don Chamberlin und Raymond F. Boyce, den Programmierern der Abfragesprache für relationale Datenbanken, Structured Query Language (SQL) zu verdanken (Boyce u. a. 1973). Im Zuge der Entwicklung der Abfragesprache SQUARE, welche Grundlage für SEQUEL, das spätere SQL, werden sollte, bemerkten die Autoren, »dass das Beispiel in Codds 1970er Artikel Tabellen sehr ähnlich sah. Wir dachten, dass der beste Weg um relationale Daten zu erklären, die Verwendung informeller Sprache wie Tabellen, Zeilen und Spalten wäre, anstatt der mathematischer Termini wie ›Relationen‹, ›Domänen‹ und ›Tupel‹. Wir entwickelten die SQUARE-Sprache mit dem Konzept der ›Tabelle‹ als Bezugspunkt« (Chamberlin 2016).<sup>xvii</sup>

Erst nachdem Codd darauf hin gewiesen wurde, fing er an, die Tabelle als Metapher für ein Operieren in mengentheoretischer Logik zu propagieren. Chris Date, ein enger Mitarbeiter und Freund Codds thematisiert dies in einem Oral History Interview. »Also sagte er [Codd] zu ihnen, was ist das grundlegendste und intuitiv verständlichste Mittel, um Information zu darzustellen? Es ist mit Sicherheit die Tabelle. Wir nutzen ständig Tabellen, für alles mögliche. Ich habe das selbst festgestellt, wenn ich mit Leuten über ihre Anforderungen an Datenbanken sprach und fragte, ›Wie sehen Deine Daten denn aus?‹ Sie beginnen Tabellen zu zeichnen, selbst wenn es IMS-Nutzer sind« (Date 2017, 23).<sup>167 xvii</sup>

---

<sup>166</sup> Ausführlicher dazu siehe *Epistemic Harvest – The electronic database as discourse and means of data production* (Hunger 2018).

<sup>167</sup> IMS, Information Management System, ist ein auf dem hierarchischen Model basierendes DBMS von IBM; Don Chamberlin bestätigt, dass Ray Boyce und er die Metapher der Tabelle erstmals als Interface für relationale Datenstrukturen in die Diskussion einführten (Chamberlin 2016, per E-Mail).

<i>supply</i>	<i>(supplier</i>	<i>part</i>	<i>project</i>	<i>quantity)</i>
1	2	5	17	
1	3	5	23	
2	3	7	9	
2	7	5	4	
4	1	1	12	

FIG. 1. A relation of degree 4

Abb. 31: Mengentheoretisch begründete Relation in Codd 1970. Abbildung in Tabellenform, jedoch ohne den Begriff Tabelle zu verwenden. Codd vermied den Begriff, den im relationalen Modell spielte im Unterschied zur Tabelle die Reihenfolge der Zeilen, bzw. der Datensätze keine Rolle (Codd/Edgar F. 1970).

Neben den oben beschriebenen Kontinuitäten gilt es nun, eine wesentliche Zäsur zu beschreiben. Während für das ›klassische‹ Tabellieren Räumlichkeit – Spalten- und Zeilenordnung – eine wesentliche Rolle spielte, änderte sich in der relationalen Tabelle deren Bedeutung. Die Relation der Zeile gewann an Gewicht, denn jede einzelne Zeile wird über einen Schlüssel bzw. eine ID-Nummer angesprochen und der Mengenalgebra zugeführt. Die Reihenfolge der Zeilen in dieser Conceptual Representation spielt keine Rolle mehr, denn Operation des Ordnen und Sortierens werden nicht mehr im Tabellenraum selbst vorgenommen und abgespeichert, sondern erst im Moment des algorithmisierten Abfragens von Informationen durch die Structured Query Language (SQL) zur Laufzeit aktualisiert. Die Abfrage bezeugt die maschinelle Agency, das Delegierende der Informationsverarbeitung. Sie ist nicht nur fragend, sondern auch strukturerzeugend, da sie neue Relationen hervorbringt. Sie formalisiert die Informationsabfrage im Zuge der Abfragesprache, die aus relationaler Algebra und Boolescher Logik besteht. Operationen, die bis dahin das ›Auge des Geistes‹ vorgenommen hatte: das Verknüpfen, Umstellen, Löschen, Hinzufügen von Informationsobjekten werden nun formalisiert und automatisiert. In der vorliegenden Diskussion ist der Schwerpunkt auf das Tabellieren in Bezug auf die elektronische Datenbank gesetzt worden. Dies ist eine andere Setzung als sie Gugerli (2012) vornimmt, wenn er ›die Abfrage‹ in den Focus setzt. Der Möglichkeitsraum der Abfrage ist nicht allein durch die Abfrage, sondern ebenso maßgeblich durch die Tabelle und genauer gesprochen, die Relation vorgegeben. Und zwar, weil die Relation(en) ein Ausdruck des Informationsmodells sind, also ein Ausdruck der Abbildung von Realität auf die algorithmisierbare Informationsverarbeitung. Kurz gesagt: Ohne Relation keine Abfrage – ohne Abfrage keine Relation. Damit deutet sich an, dass über die Rolle von SQL als Abfragesprache für durch Tabellen metaphorisierte relationale Mengen und die mit ihr einhergehenden Praktiken weiter nachzudenken ist. Eine ausführliche Untersuchung der Genealogien und Praktiken von SQL ist in dieser Studie nicht zu leisten, jedoch wird diese Fragestellung in den Kapiteln des Teil 3 immer wieder berührt werden, ohne dass dies durch ein eigenes Kapitel vertieft wird.<sup>168</sup>

Abzurunden sind diese Überlegungen zum Verhältnis von Tabellieren zum Datenbankieren zwischen Oberflächen und Tiefenschichten mit einer Spekulation: Die tiefenstrukturelle Dimension des elektronischen Rechners hat Max Bense als »Erweiterung unter die Haut der Welt« bezeichnet – der Rechner arbeitend »in den Feinstrukturen, in den Mikroverläufen der Zeit, die durch menschliches Handeln oder Denken nicht ausgenutzt werden können« (Bense 1951:436).<sup>169</sup> Diese Metapher ließe sich weiter entfalten: Psychoanalytisch gewendet, kann die Verwendung von Tabellen analog zur Struktur des

<sup>168</sup> Der Frage, wie sich die Tabelle als Interface relationaler Datenstrukturen zu anderen Strukturen wie Graphen-orientiertem Modell, XML strukturierten Daten oder unstrukturierten Datenmengen verhält, gilt es an anderer Stelle nachzugehen, um die aktuellen Wirkungsweisen des Speicherns und Operierens mit Daten aus medientheoretischer Perspektive diskutieren zu können.

<sup>169</sup> Zitiert nach (Hartmann/Frank 2015, 101).

menschlichen Begehrens interpretiert werden. Zwischenmenschliches Begehren zeichnet sich durch eine Dialektik aus Nähe und Ferne aus, wobei es der Blick, ein visueller Sinn, ist, der erlaubt, aus der Ferne das eigene (erotische) Begehren zu entwickeln. Begehren versucht seine Erfüllung zu finden, im intimen Umgang, dem gegenseitigen Ertasten, welches den Tastsinn voraussetzt und allein in der Nähe verwirklicht ist (Benthien 1999, 16f.). Dem tabellarischen Umgang, denn es stellt sich die Frage nach dem intrinsischen Vergnügen, welches mit der Tabellennutzung einhergeht, kann man eine analoge libidinöse Struktur unterstellen: Die Dialektik von Nähe und Ferne erlaubt die Projektion eines allgemeinen (Informations-)Begehrens aus der Ferne, in der Tableau-orientierten Übersicht über die dargestellten Dinge. Das Registrieren, Aus-Lesen und Operieren mit einzelnen Informationsobjekten erzeugt hingegen eine Dimension der Intimität und Nähe – ein Ertasten der Tabellenhaut.<sup>170</sup> Nähe und Ferne, Überblick und Informations-Operation, Oberfläche und Tiefenstruktur spannen den Erkenntnisraum der Tabelle auf, der durch sein Eingebettet-Sein in die alltägliche Kommunikation aus dem Blick gerät.

#### 2.2.4 Fazit

Ausgehend von der grundlegenden Bedeutung der Tabelle als visuelle Metapher für Mengen in relationalen Datenbanken sollte im Kapitel herausgearbeitet werden, auf welche Weisen das Tabellieren zum Registrieren, Delegieren und Koordinieren führt.

Erstens, ›Operationalitäten des Tabellierens‹: Für den Umgang mit Tabellen konnten anhand der historischen Beispiele ein Reihe grundlegender Operationen identifiziert werden.

- Leere Zelle: Die leere Zelle ist ein Signal, die fehlenden Informationsobjekte zu registrieren, bzw. Prozesse anzustoßen, die in Informationsobjekten resultieren.
- Zellen auslesen: Das Lesen der Zellen verbindet die räumlich abgespaltenen syntaktischen Aspekte ›Attribut‹ und ›Entität‹ zur Information und arbeitet neue Epistemen zu, beispielsweise in den mathematischen Tabellen und den Wissenstabellen.
- Referenzialität: Beziehungen unter den Informationsobjekten werden durch Operationen (Sortieren, Gruppieren, Filtern, Summe, UND, ODER etc.) aktualisiert und verändern das Verhältnis des in der Tabelle gespeicherten Informationsobjektes zu dem von ihm repräsentierten Referenzobjekt oder -vorgang. Dies erlaubt delegierende kategoriale und algorithmische Operationen, wenn beispielsweise in Transaktionstabellen Datensatzänderungen an einer Adresse, Änderungen an einem anderen Datensatz nach sich ziehen.
- Überblick: Die Blickvektoren im Tabellenraum aktualisieren die Informationspotenziale der Informationsobjekte bis die mnemotechnischen Kapazitätsgrenzen des menschlichen Gehirns erreicht sind. Die räumliche Syntax und damit die Ver-Ortung der semantischen Anteile unterliegt in der Tabelle einer beständigen Reorganisierung, die neues Wissen schafft. Die rigide, digitale, erwartbare Constructio macht Abgrenzungen und Gemeinsamkeiten maximal sichtbar und eignet sich daher für kooperative Praktiken, die der Transparenz über die Bedingungen von Information bedürfen.

---

<sup>170</sup> Zwischen dem Blick auf die Oberfläche-Haut aus der Ferne und der intimen Realisierung des Begehrens in der tastenden Nähe liegt freilich ein Moment des menschlichen Begehrens, der sich als (intellektuelle) Annäherung beschreiben lässt, und vielfach über Sprache vermittelt wird. Diesen Übergang erfassen die Metapher Benses als »Erweiterung unter die Haut« und meine Metapher des Tabellen-Begehrens nur unzureichend.

Zweitens, ›Automatisierung von Wissenspraktiken‹: In den aufgeführten historischen Tabellenpraxen unterlag die Veränderung einer Tabelle einem langwierigen Prozess des handschriftlichen Abschreibens und Überschreibens, der Hinzufügung neuer Spalten und Zeilen mithilfe von Lineal und Stift und der Vervielfältigung mit Hilfe gutenbergscher Drucktechniken. Immer war hier ein Moment der Verzögerung eingezogen, sei es, dass Summen von Hand neu berechnet werden mussten, oder alle Zeilen ausgeschnitten wurden, um sie nach einem spezifischen Kriterium, bspw. alphabetisch, neu abzulegen und gegebenenfalls zu fixieren. Tabellen in ihrer nicht-elektronischen Form wurden überschrieben, markiert und in Teilen revidiert, wobei die Änderungen sichtbare Spuren im Tabellenraum hinterließen. Diese langwierigen Verfahren der Inskription werden durch eine neue Unmittelbarkeit des automatisierten Prozessierens in Computern überholt.

So vielfältig die Tabelle historisch war, mit ihrer elektronischen Re-Operationalisierung in Tabellenkalkulationsprogrammen wie Visicalc, ab 1979, oder Excel, ab 1985 und mit ihrer Verwendung in relationalen Datenbankmanagementsystemen wie Ingres, Oracle und IBM DB2 ab Mitte bis Ende der 1980er Jahre, erreichte sie ein neues Maß an Flexibilität und Automatisierung.

Wenn man die Tabelle als simulierendes Denkwerkzeug positioniert und das Tabellieren als damit ermöglichte Wissenspraxis, so förderte die elektronische Tabellenkalkulation und die damit einhergehenden veränderten Praxen eine ständige Ad-hoc-Änderung der Parameter, die in ihren Auswirkungen – sofern die Informationsobjekte durch den Computer zeichenverarbeitbar sind – interaktiv, das heißt sofort, zur Verfügung stehen. Wiederholende manuelle Operationen von Wissensproduktion, wie Sortieren, Ordnen, Filtern, Verknüpfen wurden der Automatisierung zugeführt.

Einen Schritt weiter, in Datenbankmanagementsystemen, trifft die Automatisierung der Tabellenkalkulation auf die Möglichkeit der multiplen Verknüpfung von Informationsobjekten – sie werden »multiplizierte Multiple«. Darin liegt die Stärke von Datenbanken für Vorgänge des Delegierens und Koordinierens.

Drittens, ›Ent- und Re-Diagrammatisierung‹: Mit diesen Überlegungen gilt es, eine Unterscheidung zwischen der Verwendung der Tabelle in den Tiefenschichten des Computers, als Metapher für Relationen einzuziehen und der tabellarischen Darstellung der Abfrageergebnisse an der Oberfläche des Graphical User Interface.

- Tiefenschicht: In der Tiefenschicht des Computers spielt die Zeilenfolge keine Rolle mehr, denn die Adressierung wird automatisiert, maschinell zur Verfügung gestellt. Vielmehr ist die Relation maßgeblich.
- Oberfläche: An der Interface-Oberfläche hingegen zählt weiterhin die zur Verfügung gestellte ›klassische‹ Ordnung der Verräumlichung von Informationen die am Bildschirm oder im Ausdruck mit dem Blick abgetastet werden können.
- Vermittlung als Frage des Designs: Beide Verwendungen greifen ineinander und sind immer schon vermittelt durch Oberfläche und Tiefenschichten. Die Darstellungen an den Computer-Oberflächen sind daher nicht mehr semiotisch herleitbar, sondern sie werden zu einer Frage von Gestaltung.

Die relationale Menge ignoriert die Reihenfolge der Zeilen. Ein Lesen von oben nach unten zeigt keine Hierarchien an. Stattdessen zeigt sie, solange unsortiert, die Eingangsreihenfolge der Datensätze. Diente in der Tabelle die Abfolge der Zeilen als Adresse des Datensatzes, so ist in der relationalen Menge nunmehr die registrierende Adressierung mittels eines Schlüssels, oft einer fortlaufenden Nummer und von Fremdschlüsseln, die Datensätze in anderen Tabellen, bzw. Relationen referenzieren, realisiert.

Durch diese Abstraktion, die Loslösung der räumlichen Verteilung von der Adressierung, gelang es, die relationale Menge von der Logik sequentieller Speichermedien, die der visuellen Logik entsprach, unabhängig zu machen.

Tritt die relationale Menge schließlich als Ergebnis erneut an die Benutzeroberfläche von Datenbankfigurationen, unterliegt sie jedoch einer Re-Diagrammatisierung. Sie wird erneut in Tabellen, in Formularen oder Blöcken dargestellt, welche durch räumliche Anordnung Zugehörigkeit und Trennung und mithin Sinn erzeugen. Das Verlagern der vormals manuellen Operationen der Datenhaltung in die Tiefenschichten entzieht den Abstraktionsvorgang der unmittelbaren Anschauung und trägt so zu ›Un-sichtbarkeit‹ von Datenbankfigurationen bei, wobei die sinnliche Zugänglichmachung des Prozessierten nach wie vor im visuellen Register erfolgt.

Die Hinweise verdichten sich, dass wir mit den Abfragemöglichkeiten relationaler Mengen einer Destabilisierung der »Anschauungsform« von Epistemen beiwohnen. Diese Destabilisierung äußert sich in der relationalen Menge durch eine deutliche Steigerung der Flexibilität in epistemischen und ökonomischen Prozessen. Damit einher gehen neue Praktiken des Delegierens und Koordinierens, welche als nächstes intensiver zu untersuchen sind. Im Abschnitt 2.2.2. *Tabellarische Praktiken* stach eine Praxis heraus, welche den Übergang tabellierender, verketteter Operationen zwischen Menschen und infrastrukturellen Figurationen in eine Softwarefiguration eindrücklich darzustellen vermag: die kooperative Praxis der Transaktion. Ihr ist daher an dieser Stelle kein abschließender Abschnitt gewidmet, sondern das nächste, eigenständige Kapitel.

## 2.3. Koordination – Transaktionen

»Nur fortlaufende Identifizierungs- und Registrierungstechniken (und d. h. Medienpraktiken) gestatten Verwaltungen einerseits eine Anonymisierung und kooperative Bearbeitung der zirkulierenden, vervielfältigten und ausgewerteten Dokumente und Daten (bis zur statistischen Auswertung), und andererseits eine nachprüfbare Referenz oder ›traceability‹ der Einzelvorgänge« (Schüttpelz 2016, 13).

Anhand der vorhergehenden Kapitel haben wir gesehen, auf welche Weise relationale Datenbankfigurationen in das Registrieren von Informationsobjekten involviert sind. Dabei wurde die Tabelle als visuelle Metapher relationaler Mengen beschrieben, welche auf kulturellen Praxen der In-Formatisierung basiert, die in der Menschheitsgeschichte tief verankert sind. Das maschinisierte und automatisierte Delegieren von Informationen, mit dem Ziel Prozesse, Infrastrukturen und Logistiken aufrecht zu erhalten, ist allerdings noch nicht ausreichend beleuchtet. An welcher Stelle kippt das Delegieren um in eine andere Qualität, in das Koordinieren? Fällt beides in der Transaktion und in Datenbankpraktiken ineinander?

Die Transaktion ist ein wichtiges Beispiel dafür, wie eine spezifische Medienpraxis in der Datenbanksoftware zu Technologie wird. Sie ist nicht irgendeine Medienpraxis, sondern einer der wichtigsten medial-ökonomischen Verbünde, um die wechselseitige Abstimmung von Prozessen durch verschiedene Infrastrukturen hindurch zu garantieren und diese Garantie zu kontrollieren.<sup>171</sup>

Die Transaktion wurde durch die Informatik übersetzt, indem dem wirtschaftlichen Handeln zugrunde liegende Aktionen und Operationen algorithmisiert dem Register des Maschinischen verfügbar gemacht wurden. Dies geschieht als eine Abfolge von Operationen, die entweder vollständig oder gar nicht durchgeführt werden. Scheitert eine der Operationen, bricht die ganze Transaktion ab und die Datenbank wird auf den Zustand zu Beginn der Transaktion zurückgesetzt. Sie ist damit ein wichtiges Beispiel für die Verfung menschlicher und maschineller Agency, denn der gesamte Prozess existiert nur unter Beteiligung beider Akteure. Datenbankfigurationen ermöglichen und beschränken die koordinierende Agency der Transaktionsfigurationen. Auf welche Weise diese Einschränkung und Ermöglichung erfolgt, wird in diesem Abschnitt zu zeigen sein, es kann aber bereits angedeutet werden, dass auch hier das Informationsmodell, also die Art und Weise wie Aktionen modelliert sind, eine wesentliche Rolle spielt.

Es soll hier nicht um eine vollständig ausgearbeitete historische Darstellung gehen, sondern eher um einzelne Einsätze. Wie lässt sich eine Genealogie der Datenbank-Transaktion entwerfen? Der Begriff der Transaktion scheint auf den ersten Blick klar, ist jedoch medienhistorisch unscharf. Weder Zedler (1744) noch das Grimm'sche Deutsche Wörterbuch (1854–1961) kennen das Wort. In der sechsten Auflage von Meyers Großem Konversationslexikon wird der Begriff schließlich vermerkt:

---

<sup>171</sup> Zum Verbund aus Medien und Ökonomie schlägt Schröter in *Medien und Ökonomie – Eine Einführung* für die Medienkulturwissenschaft vor, Medien in der Schüttpelz'schen Unterscheidung von öffentlichen Medien (nicht Massenmedien) und infrastrukturellen Medien zu fassen (Schröter 2019, 13f.).

»Transaktion (lat.), Verhandlung; Unterhandlung zur Beilegung von Streitigkeiten; Vergleich, Übereinkunft; auch Handelsunternehmung« (1888, 796). Der merkantilen Bedeutung des Begriffes – einer gegenseitige Übertragung von Waren und Dienstleistungen – geht jedoch eine kommunikative voraus. ›Transaktion‹ trat wissensorientiert als Titel von Veröffentlichungen gelehrter Gesellschaften auf, prominent in den *Philosophical Transactions of the Royal Society*, die ab 1665 erschienen. Wie also lässt sich mit diesen disparaten Einträgen umgehen?

Die vorliegenden lexikalischen Einträge helfen nur bedingt weiter. Es ist daher ein anderer Weg einzuschlagen, der allerdings keine vollständige historische Rekonstruktion unternimmt, sondern eine einzelstehende Praxis der Transaktion beispielhaft herausarbeitet. Der vorliegende Abschnitt nimmt eine medial-ökonomische Operationskette der Informationsverarbeitung in den Blick: die doppelte Buchhaltung. Anhand dieser im 15. Jahrhundert emergenten Verwaltungspraxis wird aufgezeigt, wie das Registrieren von Vorgängen als Information neues Wissen schafft, und wie diese mediale Praxis soziale und ökonomische Verhältnisse verändert. Damit wird das Einsetzen transaktionaler Praxen bereits wesentlich früher markiert, als dies beispielsweise durch Lauer geschieht, welcher Transaktionen anhand der Einführung von Kreditkarten in Nordamerika ab 1888 einordnet (Lauer 2020). Konkret wurde in Europa ab dem 15. Jahrhundert mit Hilfe der doppelten Buchhaltung neue semantische Verknüpfungen zwischen einer zeitlichen Logik (der Transaktionen) und einer Sachlogik (der Gegenstände der Transaktionen) geschaffen. Wie diese damals neuen Medien der Ökonomie mit sich verändernden Wissenspraxen einherging, ist im ersten Abschnitt 2.3.1 *Der Medial-Ökonomische Verbund* aufzuzeigen.

Nachdem diese erste Spur gelegt ist, wird ein zweiter genealogischer Zweig aufgerufen, indem detailliert beschrieben wird, wie die manuellen Transaktionspraxen mit elektro-mechanischen Geräten ergänzt und automatisiert wurden, und auf welche Weise diese Automatisierung von Transaktionspraxen den Weg ebnete für die Computerisierung und sukzessive Übersetzung in jene Softwarefiguration, die als Datenbankmanagementsystem bekannt wurde. Dem widmet sich der Abschnitt 2.3.5 *Ein Ticketverkäufer wird zum Informationsanbieter*

Die weiter zu beschreibenden Übergänge zur Transaktionsabwicklung erfolgten durch den nicht-linearen Speicherzugriff und mit elektronischen Datenbanken, welche Zugriffe mehrerer Nutzer gleichzeitig ermöglichten. Dazu zählen auch Ansätze zur Standardisierung und Formalisierung, wie die Transaktionskontrolle durch Control Spheres und das ACID-Modell. Dies wird in Abschnitt 2.3.6 *On-Line Transaction Processing (OLTP)* ausgeführt.

Eine Dimension ist die Einführung von Datenwerkzeugen, die es erlaubten, große Mengen von Daten zu analysieren und in den 1990er unter dem Begriff On-Line Analytical Processing (OLAP) zusammengefasst wurden. Sie ebneten die Wege bis in die Jetztzeit, in der mit Big-Data ›streng‹ transaktionsorientierte Ansätze wie das CODASYL-Netzwerkmodell und die relationale Algebra durch ›nachlässigere‹ Post-SQL Figurationen abgelöst werden. Heute orientieren sich Geschäftsmodelle, verkörpert durch Firmen wie Facebook, Amazon und SAP, an der Verwertung von *Analysedaten* von Transaktionen. Erarbeitet werden diese Punkte im Abschnitt 2.3.7 *On-Line Analytical Processing (OLAP)*.

### 2.3.1 Der Medial-Ökonomische Verbund der doppelten Buchhaltung

Wenn wir zurückblicken auf Momente in der Vergangenheit, in welchen sich heutige ökonomische Organisationsweisen herausgebildet haben, führt der Blick nach Venedig und Milan, Ende des 15.

Jahrhundert.<sup>172</sup> Organisation und Verwaltung des Handels unterlagen in dieser Phase deutlichen Veränderungen, die im Folgenden anhand der doppelten Buchhaltung skizziert werden sollen.<sup>173</sup>

Dabei soll die Buchhaltung des 15. Jahrhunderts nicht transhistorisch mit der Buchhaltung des 20. und 21. Jahrhunderts gleichgesetzt werden, denn es unterschieden sich nicht nur die medialen Niveaus der Gesellschaften, sondern auch deren ökonomische, kulturellen und sozialen Verfassungen. Statt einer transhistorischen Gleichsetzung geht es vielmehr um eine Untersuchung der Tiefenzeiten von Medien (McPhee/Zielinski).

Die doppelte Buchhaltung war zwischen 1300 und 1500 eine sich herauschälende neue medial-ökonomische Praxis mit Informationen und steht stellvertretend für eine neue kooperative Praxis des Handels, welche durch Kapital vermittelt wurde. Der Handel zeichnete sich außerdem in dieser Periode durch die zunehmende Diversifizierung der Handelstätigkeiten aus. So fingen Händler an, neben reinen Handelsgeschäften zusätzlich auch Geld zu verleihen. Wie konnten sie angesichts der komplizierter werdenden Transaktionen sicher sein, dass ihre Bücher, ihre Buchhaltung, stimmig blieben? Wie korrespondierten die neuen Techniken der Verwaltung, mit der nunmehr depersonalisierten Verfügbarkeit von Geld durch Kredite und Anleihen?

Eine hier zu entwerfende medienhistorische Fluchtlinie besteht darin, die doppelte Buchhaltung als historisch emergente epistemische Form zu rekonstruieren, in deren Zuge die doppelte Buchhaltung, ebenso wie die zuvor beschriebene Tabelle, als Werkzeuge der Wissenskonstitution zur Verfügung stehen. Denn seit der Renaissance unterlagen die epistemischen Methoden in Europa dramatischen Veränderungen in einer Bewegung. Diese Bewegung führte vom Erfassen der metaphysischen Essenz eines Gegenstandes zu einem Wissensmodell des Fakts und als Steigerungsform, zu quantifizierbaren Fakten, wie die Historikerin Mary Poovey ausführte. Sie argumentiert, dass es nicht die Mathematik allein war, die als Wissenschaft betrieben, der Etablierung ›des Faktischen‹ in den Wissenschaften Vorschub leistete. Vielmehr, so Poovey habe die angewandte Arithmetik im algorithmischen Gewand der doppelten Buchhaltung den Weg für die Entwicklungen des 19. Jahrhunderts geebnet (Poovey 1998, 29). Mit

---

<sup>172</sup> Diese Zeitangabe bezieht sich auf die weitreichende Verbreitung in Europa, da sie argumentativ auf die Frühphase des Kapitalismus und die in Europa ab dem 19. Jahrhundert prosperierende Industrialisierung abzielt. Aktuelle Forschungen zeigen, dass ähnliche Praktiken bereits in Alexandria und Konstantinopel in der Frühzeit der ägyptischen Mamluken-Dynastie (1250–1517) Verwendung fanden. Das älteste, noch erhaltene Dokument doppelter Buchhaltung in Europa, datiert auf 1340, ist das Genoveser Massari-Buch, welches die städtischen Finanzen listete (Lauwers/Willekens 1994, 299f.).

<sup>173</sup> Der Zuspitzung auf die Transaktion ist in eine Vielzahl bürokratischer Praxen einzuordnen, die umfassend beforscht wurden. Die soziologischen Ansätze von Weber (1921/22) und weiteren sollen an dieser Stelle nicht ausgeführt werden. Medientheoretisch bedeutend sind eine Reihe von Publikationen, die ab Mitte der 1990er Jahre anhand von Fallbeispielen eine Archäologie des Bürokratischen betrieben: Joanne Yates untersuchte die Emergenz kontrollierender bürokratischer Praktiken und Techniken in drei amerikanischen Unternehmen zwischen 1850 und 1920 in *Control through Communication – The Rise of System in American Management* (Yates 1993). In dem von Vogl/Siegert 2003 herausgegebenen Sammelband *Europa – Kultur der Sekretäre* untersuchen medientheoretische Einzelbeiträge ›Sekretäre‹ als Verwaltungsinstrument, Schreibzeug, Arbeitsverhältnis und Schauplatz von Macht (Siegert/Vogl (Hrsg.) 2003). Claudia Vißmann erkundete in *Files – Law and Media Technology* die Akte als bürokratische Praxis der Rechtsordnung (Vismann 2008). Krajewski ging in *Paper Machines – about Cards & Catalogs, 1548-1929* (Krajewski 2011) den bibliothekarischen Ordnungssystemen der Karteikarten und Kataloge in ihren gegenseitigen Übergängen nach. Der Sammelband *Medien der Bürokratie* untersuchte die kleinteiligen Ermöglichungsbedingungen von Bürokratie (Balke/Vogl/Siegert (Hrsg.) 2016).



dieser Argumentation soll untersucht werden, wie die doppelte Buchhaltung eine neue Form von Wissen quasi nebenher ›abwarf‹ – jenes Wissen, dass durch die Verzeichnung von Transaktionen, zusätzlich entstand.



Abb. 32: Jacopo de Barbari: Ritratto di Fra Luca Pacioli, ca. 1495 (Barbari/de 1495).

Das dem italienischen Renaissance-Maler Jacopo de Barbari zugeschriebene Bildnis Luca Paciolis versammelt eine Reihe von Motiven, die für das Leben des Franziskanermönches Luca Paciolis bestimmend waren (Abb. 32). Pacioli verbrachte seine Jugend und erhielt seine Grundbildung im Kloster San Sepolchro (Toskana), wo er sich mit dem Maler und Mathematiker Piero della Francesca anfreundete, der ihm Zugang zu, für damalige Zeiten, ungeheuren Bildungsreichtümern verschaffte. Neben seinem Studium an der Universität von Padua war ab er 1464 als Hauslehrer für die Söhne und als Buchhalter der Händlerfamilie Rompiasi tätig. Teilweise agierte er im Auftrag der Familie Rompiasi selbst als Händler. Spätestens hier erhielt er Einblick in die Buchhaltung ›nach Art der Venezianer‹: die doppelte Buchhaltung. Während seines Aufenthaltes in Urbino beendete er sein erstes Hauptwerk *Summa de arithmetica, geometria, proportioni et proportionalita*, welches am 20. November 1494 in Venedig veröffentlicht wurde (Mackinnon 1993, 154). Ein Jahr später entstand das Gemälde de Barbaris. Heute würde man die *Summa* als Kompilation beschreiben: Bis zu drei Viertel gaben die Arbeit Fibonacci, Euklids und Piero della Francesca's Manuskripte *Trattato d'abaco* und *De quinque corporibus*

*regularibus* wieder – womit sich Pacioli in der mittelalterlichen Tradition des Kopierens von Texten bewegte (Lauwers/Willekens 1994, 292).<sup>174</sup>

Die Bedeutung der neuen Methode wird deutlicher vor dem Hintergrund damals üblichen Buchhaltungstechniken. Ursprünglich ging es im Rahmen der, an die Famiglia gebunden, ökonomischen Aktivitäten um zu hütende Geheimnisse des Wirtschaftens. Buchhaltung war keine spezifisch auf Geschäftsvorgänge konzentrierte Technik, sondern vermischte sich mit formalen und nicht-formalen Dokumenten der jeweiligen Famiglia. Kontenlisten, Verkaufsbelege, Schuldscheine wurden mit Gebetsbüchern, Erinnerungen, Erbstücken, und Verweisen auf die Familiengeschichte vermischt aufbewahrt. Diese Sammlungen ähnelten, so Poovey, Tagebüchern stärker als Kontenbüchern (Poovey 1998, 34).<sup>xcviii</sup> Im Kapitel *Particularis de Computis et Scripturis* (dt. *Details des Rechnens und Aufzeichnens*) verschriftlichte Pacioli die Kulturtechnik der doppelten Buchhaltung, die sich in den Städten Genua, Venedig und Florenz ab dem 13. Jahrhundert herausgebildet hatte. Historiker vermuten, dass dieses Wissen durch italienische Händler aus Alexandria und Konstantinopel oder anderen Städten Ägyptens und Syriens importiert wurde (Goody 1977, 89). Die buchhalterische Praxis basierte auf drei Stützen:

- den allgemeinen Tageseinträgen in das Memorandum,
- den Journalen, in welchen einzelne Transaktionen aufgezeichnet wurden
- und dem Sachkontenbuch beziehungsweise Register, welches als zentral angelegt und mit einem Index versehen war.<sup>175</sup>

#### Inventar

Position	Soll	Haben
Flour sold July, 7		6 pound
Flour bought July, 20	10 pound	

#### Konto

Position	Soll	Haben
Flour sold July, 7	2 ducats	
Flour bought July, 20		30 ducats

Abb. 33: Prinzip der doppelten Buchhaltung. Oben: Sachkonto, unten Buchungskonto (Autor).

Jede Transaktion soll zweimal niedergelegt werden. Ein Betrag, der auf einem spezifischen Konto als Soll verbucht wurde, musste auf einem anderen Konto als Haben verzeichnet werden. Wenn zum Beispiel 6 Pfund Mehl für 2 Dukaten verkauft wurden, so musste sechs Pfund Mehl vom Konto ›Inventar‹ abgezogen werden und im Kassenbuch 2 Dukaten als Einkunft verzeichnet werden. Ein und dieselbe Transaktion wurde doppelt niedergelegt (Abb. 33). Dabei erfolgte eine semantische Aufteilung der Einträge nach einer zeitlichen Logik des Ein- und Ausgangs (im Kassenbuch) und nach einer sachlichen Logik, der Zuordnung zu einem Sachgebiet (die Zuschreibung zum Konto Inventar). Verbunden mit dieser semantischen Aufteilung war die Idee des Belegs. Sowohl die Transaktion im Kauf/Verkauf als auch das aktuelle Inventar wurden beleghaft nachweisbar. Damit gelang es, *jederzeit* eine Bilanz der

<sup>174</sup> Zur mittelalterlichen Wissenskonfiguration und der Entwicklung des Buchdrucks siehe *Die Gutenberg-Galaxis – Das Ende des Buchzeitalters* (McLuhan 1962). Die ›andere‹ Seite der Buchhaltung, die Lagerhaltung ordnet Monika Domman in *Wertspeicher – Epistemologien des Warenlagers* (2012) ein.

<sup>175</sup> Ausführlich siehe *Five Hundred Years of Bookkeeping – a Portrait of Luca Pacioli* (Lauwers/Willekens 1994, 296–299).

Geschäftstätigkeiten aufstellen zu können, die zum Beispiel für die Aufnahme von Krediten den Gläubigern die Vermögenslage glaubhaft darzustellen erlaubte.

Das ›doppelte‹ der Buchhaltung bezieht sich auf die gleichzeitige doppelte Niederlegung der Transaktion (Geijsbeek 1914). Die Transaktion wurde als Gegenstand kaufmännischen Denkens explizit gemacht und zu Datum erhoben. Ein Datum, das aufgezeichnet werden konnte und einen Erkenntniswert mit sich brachte. Die Gewinnung von Transaktionsdaten tritt hier als eine Operationskette auf, welche einen epistemischen Mehrwert generiert (Fischer 2000, 299). Die Bilanzen und Profite wurden nicht erst, wie in der einfachen Buchhaltung, nach Abschluss eines Buches oder nach Ablauf eines Kontrakts sichtbar, sondern in deutlich kürzeren Intervallen – im Idealfall: täglich. Darin lag das Neue der doppelten Buchhaltung.

Die Einführung der doppelten Buchhaltung war in eine Reihe von Informationspraktiken des Warenhandels eingebettet. So begründete sich der Erfolg von reisenden Warenhändlern im Mittelalter in einem Informationsvorsprung: Da das Wissen über verschiedene Preise nicht zwischen den Märkten zirkulierte, konnten sie an einem Ort günstig einkaufen und an einem anderen Ort teuer verkaufen. Dieser Informationsvorteil schwand mit der zunehmenden Reisetätigkeit zwischen den Marktplätzen, ermöglicht durch Straßen- und Seetransport-Infrastruktur und mit zunehmender Verbreitung allgemeinen Wissens (z.B. Sprachen), denen ein steigender Informationsaustausch folgte. Dass sich die Möglichkeiten zu Gewinnen aus Informationsunterschieden verringerten, führte dazu, dass die Händler auf andere Gewinnmöglichkeiten ausweichen mussten. Diese bestanden beispielsweise in der Einfuhr veredelter Rohstoffe, deren sukzessiver Veredelung und der anschließenden Veräußerung mit Gewinnen, wie beim Tee-Import (Kindleberger 1975).

Die gedruckte Veröffentlichung der damals bereits bekannten Methode wurde deswegen so wirkmächtig, weil der Buchdruck – an dieser Stelle verschränken sich Wirtschaftsgeschichte und Mediengeschichte – im kurz zuvor erfundenen Gutenbergschen Letterndruck erfolgte, der eine hohe Vervielfältigung ermöglichte. Paciolis Werk wurde in den folgenden Jahren in Europa rezipiert, kopiert und weiter verbreitet, so in Hugh Oldcastles *Profitable Treatyce*, London, 1543, Wolfgang Schweickers *Zwiffach Buchhalten*, Nürnberg, 1549, Ausgaben in den Niederlanden 1543 und Frankreich 1543 (Lauwers/Willekens 1994, 301). Dies bedeutet jedoch nicht, dass die doppelte Buchhaltung Mitte des 16. Jahrhunderts als vollständig durchgesetzt gelten kann. Neben dieser Form war bis in das 20. Jahrhundert hinein die einfache Buchhaltung und andere Formen weiterhin in Anwendung.

### 2.3.2 Die Depersonalisierung des Kapitals und dessen Verzeichnung

»Was nun aber der doppelten Buchführung jene entscheidende Bedeutung für die Entwicklung kapitalistischen Wesens verleiht, ist der Umstand, dass sie in der That erst in ganzer Vollendung, die der kapitalistischen Geschäftsführung entsprechende Systematisierungen ermöglicht« (Sombart 1902, 1:393f.)

Nun gilt es zu zeigen, wie die Operationsketten des doppelten Buchhalten ineinander fielen mit einem ökonomischen Prozess, in dem Kapital von den ursprünglichen Verwendungszusammenhängen zunehmend abstrahiert wurde. Damit lässt sich argumentieren, dass neue oder emergente Medienpraxen, welche neue Wissenspraxen mit sich bringen, eng mit ökonomischen Veränderungen einher gehen.

Die Vergesellschaftung des Kapitals wird als wichtige Voraussetzung für den langen Übergang von einer mittelalterlich-merkantilen zu einer modern-industriellen Ökonomie beschrieben. Während bis zum Einsetzen dieses Prozesses Vermögen an Individuen gebunden waren und sich in persönlichen Verhältnissen widerspiegeln, verlor vergesellschaftetes Kapital seine ›Identität‹. Es wurde de-personalisiert: ›Kapital-an-sich‹. Kapital war nicht länger gebunden an Familienverbände, oder einzelne Personen, sondern wurde in Gesellschaften zusammengelegt (ebd.). Hinter dem Rücken der Beteiligten etablierte sich ein abstraktes gesellschaftliches Prinzip: Zweck des Warentauschs war nicht länger die unmittelbare Bedürfnisbefriedigung, sondern die Vermehrung von Kapital-an-sich. Auf die Händler bezogen bedeutete dies den Einsatz von Kapital zum Ankauf von Waren, nicht als Vorgang der Versorgung, sondern orientiert an dem Ziel, diese mit Gewinn zu veräußern. Beispielhaft war dies im Zuge der Handels-Revolution in den nord-italienischen Städten ab dem 13. Jahrhundert zu beobachten. So waren zuvor Partnerschaften, typischerweise als *Fraterna*, als Vertrag zwischen Brüdern, organisiert, welche gemeinsam ihr Erbe verwalteten und in Handel investierten – eine personalisierte, an Individuen gebundene, Form des Vermögens.<sup>176</sup> Dies begrenzte jedoch den Zufluss weiteren Kapitals für die Ausweitung von Handelstätigkeiten, denn die Vermögensausweitung beispielsweise durch Einheirat war im Prinzip nur einmal möglich. Für den Einsatz zusätzlichen Kapitals reichten die familiären Bande nicht aus. Dies war nur durch den Zugriff auf Einzelkapitale Dritter zu gewährleisten. Das beförderte die Entstehung der erweiterten *Compagnia*, in welche Vermögen Dritter als Kapital de-personalisiert einfließen konnte (vgl. Hunt 1994, 12). Das Kapital wurde vergesellschaftet, es begann als Kapital-an-sich zu zirkulieren. Ebenfalls neu, erlaubte das Auftreten von Händler-Banken und Buchgeschäften die Zirkulation von Geld jenseits personalisierter Beziehungen (Bryer 1993, 120f. und 135).

In einer Einführung zur *Kritik der politischen Ökonomie* fasst Michael Heinrich 2005 zusammen:

»Zweck des Prozesses ist jetzt die quantitative Vermehrung der ursprünglichen Geldsumme. Das Geld wird nicht verausgabt, sondern vorgeschossen; es wird nur ausgegeben, damit anschließend mehr eingenommen wird. Eine Wertsumme, die *diese* Bewegung vollzieht, ist Kapital. Eine bloße Wertsumme für sich genommen, sei es nun in Gestalt von Geld oder in Gestalt von Waren, ist noch nicht Kapital. Auch ein einzelner Austauschprozess macht aus einer Wertsumme noch nicht Kapital. Erst die Verketten von Austauschprozessen mit dem Zweck, die ursprüngliche Wertsumme zu vermehren, liefert die typische *Kapitalbewegung*: Kapital ist nicht einfach Wert, sondern *sich verwertender Wert* [...]«

(Heinrich 2005, 83).<sup>177</sup> Die Austauschprozesse nahmen einen höheren Grad an Komplexität an, der erst durch neue Praktiken des Registrierens, Delegierens und Koordinierens Sichtbarkeit erlangte.

Es ließe sich argumentieren, dass das Aufzeichnen von Inventaren, und Geschäftsvorfällen eine jener kulturellen Techniken ist, die selbst dem Alphabet vorhergehen, da Inventar bereits in den Hieroglyphen-Tontafeln der Sumerer 2500 v.u.Z. niedergelegt war. Dabei bezieht sich das hier entwickelte Argument nicht auf den Umstand des Aufzeichnens an sich, sondern auf die doppelte Verzeichnung, welche die zwei Seiten – einem Zugang steht immer ein Abgang gegenüber – ein und desselben Vorgangs überhaupt erst produziert. Diese zwei Seiten existieren voneinander nicht unabhängig, sondern stehen in ständiger Beziehung. Im abgebildeten Buch der J. and H. Hadden and Company Limited Nottingham ist diese Beziehung diagrammatisch gelöst, durch Verzeichnung in der gleichen Zeile und Hervorhebung mittels Unterstreichung der zueinander gehörigen Summen (Abb. 34).

<sup>176</sup> Der Ausschluss von Frauen ist hier kein historischer Zufall, sondern Teil des historischen Patriarchats.

<sup>177</sup> In seiner Argumentation weicht Heinrich von Bryers Interpretation ab, dies auszuführen, ginge hier zu weit (vgl. Heinrich 2005, 78–82).

Eine andere Möglichkeit, neben der diagrammatischen Lösung, die zwei Seiten des Soll und Habens in Beziehung zu halten, war die Verwendung von Schlüsselnummern. Diese können als fortlaufende Nummerierung auftreten oder komplexerer Natur sein, wie Signaturen in Bibliotheken. Die Verwendung von Schlüsselnummern fügt den einzelnen Datensätzen eine zusätzliche Qualität hinzu, sie erhalten eine Adresse. »Adressen sind Daten unter denen andere Daten überhaupt erst erscheinen können« so Kittler (1988, 191). Dazu Gloria Meynen: »Jede Buchung kann genauestens adressiert werden. Jede Kasse findet ihr Kapital, jedes Soll sein Haben. Die Einträge des Hauptbuches verweisen mit Seitenzahl aufeinander. Sie setzen Zeiger von der linken auf die rechte Seite, von der rechten Seite auf die linke Seite der Tabelle« (Meynen 2004, 187).

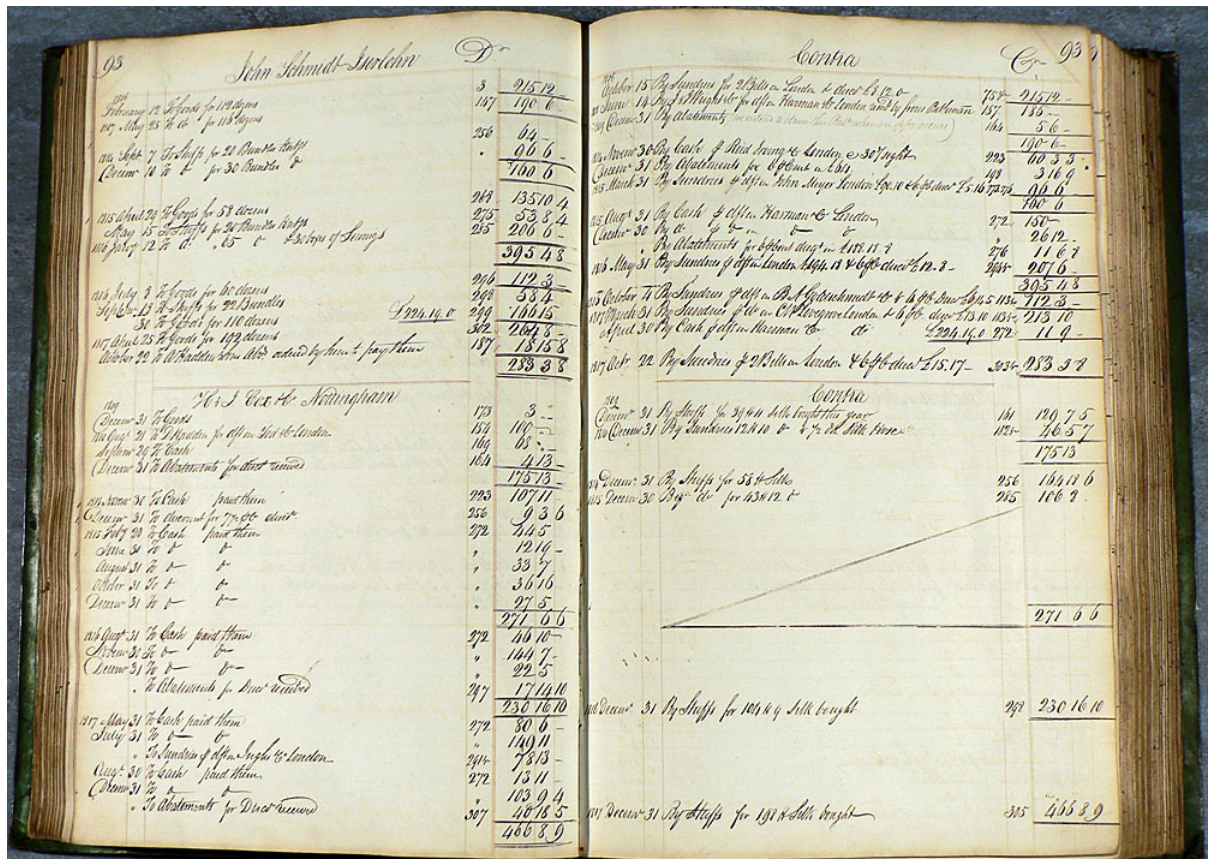


Abb. 34: Double-entry ledger from J. and H. Hadden and Company Limited, Hosiers, Nottingham 1685-1915. Links: D – Debit (Soll), Rechts C – Credit (Haben). Die unterstrichenen Summen auf beiden Seiten des Buches sind gleich (University of Nottingham, Ha A 3-5).

Adressen registrieren nicht allein die in den Büchern doppelt niedergelegten Transaktionen, sondern auch die Belege, die als Beweise für den Wahrheitsgehalt der Verzeichnung dienen. Die fortlaufende Nummerierung erzeugte neue epistemische Netzwerke einerseits und Zwänge andererseits. Sie machte Leerstellen oder fehlerhafte Einträge sichtbar und hielt zur Präzision der Verzeichnung an, unter Androhung des visuellen Makels von Streichungen, Überschreibungen und anderen sichtbaren Korrekturen.

Die wichtigste Aufgabe der Adressierung in diesem Geflecht: Sie verknüpft dauerhaft zwei semantische Logiken: Die zeitliche und die sachliche Logik sind noch Jahre später als Beziehung auffindbar und rekonstruierbar.

Gleichzeitig, wie Poovey argumentiert, verschob sich der Status der Autoren. Autorenschaft war nicht länger an die Zugehörigkeit zur *Fraterna* gebunden. Sie war nicht länger allein dem Patriarchen vorbehalten. Sie verkündete nicht länger den Status der Autoren, wie es noch in den losen Sammlungen der vormodernen Zeit der Fall war, und »die Autoren wurden austauschbar« (Poovey 1998, 35).<sup>xciix</sup> Damit wirkt die doppelte Buchhaltung als Kontrollfiguration »zivilisierend« nicht allein auf den Familienvorstand, sondern auf alle Beteiligten, auch die Beauftragten und Angestellten. Diese kulturtechnische Verfügung ermöglichte eine Ausweitung des Personals, denn die Figuration der doppelten Buchlegung war um vieles transparenter, als ihre Vorgängerinnen. Sie verschob den Diskurs des Ökonomischen aus dem privaten Raum der *Famiglia*, in die sich in der frühmodernen Renaissance konstituierende Öffentlichkeit.<sup>178</sup> Das Wissen über den Zustand des eigenen Unternehmens, über welches ein bestimmter Händler implizit verfügte und welches in seiner Abwesenheit oder nach seinem Tod nicht mehr verfügbar war, wurde expliziert und löste sich von der Person des Händlers ab. Diese Transparenz fügte sich ein in den Prozess der De-Personalisierung von Kapital – personenunabhängig konnte der Zustand eines Unternehmens bestimmt werden, allein anhand der vorliegenden Bücher.

Rob A. Bryer verschränkt die Frage der Kapitalbewegung mit dem Aufkommen der doppelten Buchführung, indem er darstellt, dass erst die doppelte Buchhaltung überhaupt erlaube, die Rentabilität fortlaufend zu ermitteln. »Jede Transaktion kann auch in Hinsicht auf ihre Auswirkungen auf die Rentabilität (Gewinn geteilt durch Kapitaleinsatz) gewertet werden. [...] Für Händler im kapitalistischen Sinne ist die *Rentabilität* ausschlaggebend, und nicht allein Gewinn« (Bryer 1993, 114f.).<sup>c</sup> In Bryers Hinweis auf die Rentabilität drückt sich aus, dass die Einzelkapitalien miteinander in beständiger Konkurrenz bei Strafandrohung des eigenen Untergangs stehen. Daher ist die Gewinnerzielungsabsicht allein für das Argument nicht ausschlaggebend, sondern der Umstand, das eigene Unternehmen in die Lage zu versetzen, andere Kapitalien niederzuzukonkurrieren. Bryer geht der These nach, dass die doppelte Buchhaltung entstand, »als das Kapital vergesellschaftet wurde, in Reaktion auf den gemeinsamen Bedarf von Investoren nach einer *regelmäßigen* Berechnung der Rentabilität, welche die Grundlage für das Aufteilen der Gewinne darstellt« (ebd., 115).<sup>ci</sup> Zur Durchsetzung der doppelten gegenüber der einfachen Buchhaltung kommentiert er, dass zwar auch die einfache Buchführung erlaube, Gewinn und Vermögen darzustellen, dies jedoch nicht automatisiert erfolge. Während es viele Formen der einfachen Buchführung gäbe, sei die doppelte Buchführung ein »einzigartig spezifizierter Algorithmus« (ebd.).<sup>cii</sup> Was Bryer als regelmäßige Berechnung der Rentabilität konkretisiert, soll hier als die Generierung epistemischen Wertes, eingebettet in eine mediale Praxis, gekennzeichnet werden.

Das Spezifische der sich verbreitenden doppelten Buchhaltung lag in ihre inhärent kontinuierlichen Produktion von Daten (in Bezug auf Gewinn und Vermögen). Jede Transaktion wurde verzeichnet und überhaupt erst als solche sichtbar. Im Unterschied dazu, war die einfache Buchhaltung periodisch angelegt: sie ließ diese Daten in größeren Zeiträumen sichtbar werden und nicht ad hoc. Die doppelte Buchhaltung wurde zu einer Methode, die es erlaubte, aus formatierten Informationen Wissen zu generieren – ein spezifisches Wissen, das es potentiell ermöglichte, einen Vorsprung vor der Konkurrenz zu erlangen – und einen epistemischer Mehrwert schuf.

---

<sup>178</sup> Wobei Öffentlichkeit hier einzugrenzen ist auf jene, die an dieser Form der Öffentlichkeit teilnehmen durften und jene, die davon ausgeschlossen waren: Frauen, Kinder und andere Subordinierte.

### 2.3.3 Verschränkungen kaufmännischer und statistischer Wissensfelder

In der Neuzeit verschränkten sich die kaufmännischen Wissenspraktiken mit jenen des Regierens.

Wie kam es zu dieser Übertragungsleistung?

Die buchhalterische Praxis der Kaufleute entwickelte ein tiefgehendes Methodenwissen, welches sich mit jenen mediale Praktiken der formatierten Informationsverarbeitung vermählte, wie sie in Bibliotheken gepflegt wurden – es galt mit den wachsenden Informationsmengen die Frage ihrer Archivierung und des Zugriffs zu beantworten. »Alles Inventar aufzuzeichnen, war ein umfangreiches Vorhaben: Häuser, Land, Lager, Produktionsmittel, Bargeld, Zahlbelege und Schuldscheine. Das Inventar für ein großes Haus allein konnte zahlreiche Aktenbände umfassen. Ein Händler musste um diese Information zu verwenden, die Indexierungs-Werkzeuge und die Diskurse, die für das Aufzeichnen, Summieren, und Verständnis der Kontenbewegungen nötig sind, beherrschen« (Soll 2009, 56).<sup>ciii</sup>

<b>Les remboursements faits par le Roy de ses domaines, droits et revenus aliénés depuis et avant son avènement à la couronne, montoient, suivant le détail rapporté en l'abrégé de l'année 1672, à.....</b>	
	<b>170,277,982<sup>1</sup></b>
En l'année 1673, à.....	2,084,720
En l'année 1674, à.....	4,498,238
En l'année 1675, à.....	1,206,799
En l'année 1676, à.....	884,166
En l'année 1677, à.....	1,450,958
En l'année 1678, à.....	758,292
En l'année 1679, à.....	34,572,398
En l'année 1680, à.....	10,792,927
<b>TOTAL 2.....</b>	<b>226,526,482 226,526,482<sup>1</sup></b>

Abb. 35: Staatsstatistik: Einnahmen des Königs aus Gütern, Rechten und Veräußerungen in den Jahren bis 1680, zusammengestellt durch Colbert im Abrégé des Königs (Colbert 1865, II/2:771).

Beispielhaft lässt sich diese Bewegung am Königshof Ludwig des XIV. nachzeichnen. In der Person des Finanzministers Jean-Baptiste Colbert verschränkten sich beide Praxisfelder, Bibliothekswissen und Buchhaltungswissen, und formten die Grundlage neuer gouvernementaler Praxen, der Polizeywissenschaften des 17. Jahrhunderts.<sup>179</sup> Colbert hatte in seiner Familie eine Ausbildung als Tuchhändler genossen und stieg zum engsten wirtschaftlichen Berater des Königs auf. Er überarbeitete und forcierte nicht nur die statistische Erfassung der Luxusgüterproduktion des Königs, er erhob diese neue epistemische Methode zur Basis für die Regierungsentscheidungen des Königs, womit – so kann man argumentieren – sich die Rolle des Herrschers wandelte. War es bis dahin die Aufgabe als Herrscher, das Recht zu kennen und das Recht zu sprechen, sollte der König nun vielmehr seinen Staat, das heißt seine Bevölkerung, kennen – um diese zu regieren. Statt ›Ich bin das Recht‹ hieß es nun ›Ich bin der Staat‹. Die Rolle des Königs wandelte sich zu einem verwaltenden Regenten.<sup>180</sup> Die Folianten Colberts, in die der König Einsicht nahm und auf die er mit Anweisungen reagierte, repräsentieren diesen Wandel und verwirklichen sich in der tabellaren, sich ausbreitenden Übersicht über die zu regierenden Dinge.

Der Souverän sollte die Bedürfnisse seines Volkes kennen, um es angemessen regieren zu können. Er musste die wirtschaftliche Lage kennen und über außenpolitische Belange informiert sein. Die ›Kunst des Regierens‹ – die Berater der französischen Könige hatten bis Colbert einen humanistischen oder

<sup>179</sup> Ausführlich siehe *The Information Master – Jean-Baptiste Colbert's secret state intelligence system* (Soll 2009, 50ff.).

<sup>180</sup> Damit wurde langfristig die Ablösung der Regierungstätigkeit von der Person des Königs denkbar; und zwei Regentschaften später unter Ludwig XVI., mit der französischen Revolution von 1789 auch Realität.

kirchlichen, jedoch keinen merkantilen Hintergrund – wandelte sich zur ›Technik des Regierens‹. Einher ging dies mit neuen Werkzeugen der Informationserhebung und -verarbeitung. Colbert führte Buch für den König, er brachte ihm die Kunst der doppelten Buchhaltung nach Pacioli bei. Ein fast ungeheurer Vorgang: Der Sonnenkönig erwarb wirtschaftliches Methodenwissen, um die Buchführung zu verstehen und direkt zu kontrollieren (Abb. 35).

Damit zieht eine neue Ebene in das statistisch begründete Regierungshandeln ein, die nicht allein die Abbildung von Realität, sondern in stärkerem Maße noch deren Modellierung vergegenwärtigt. Die doppelte Buchhaltung in Tabellen und die Gewinnung von Transaktionsdaten trägt zur Modellierung des Staates bei – ein Argument, welches ähnlich in Leibniz' Entwurf gewisser Staatstafeln anzutreffen ist (vgl. Leibniz 1680).

Seine Privat-Bibliothek erlaubte es Colbert, aus der Perspektive des Kaufmanns Wissen anzusammeln, das neu formiert zu ökonomischen Staatswissen wurde. Diese Indienstnahme des Wissens ist nicht abzutrennen von einer universellen Wissensproduktion der Bibliotheken, Museen und Sammlungen seit dem 17. Jahrhundert. Für Colbert galt es, Wissen über andere Nationalstaaten zu registrieren und produktiv zu machen, um wirtschaftliche Entscheidungen auf nationaler Ebene begründen können. Eine dieser Analysen führt beispielsweise zum gezielten Ausbau der Luxus-Produktion von Seide in Frankreich. Skizziert sehen wir in diesen Bemühungen die Verflechtung von spezifischem wirtschaftlichen Methodenwissen und jenen Methoden der Wissensorganisation ästhetischer und gesellschaftlicher Felder, die sich in der Frühneuzeit entwickeln.<sup>181</sup>

Aus der Warte des Staates verschob sich die Aufmerksamkeit zwischen den Wissensfeldern. Indem die Könige »nach und nach überzeugt wurden, dass Zahlen nutzbringend sein konnten, weil sie präzise Aufzeichnungen von Geld erzeugen und daher nationalen Reichtum indizieren konnten, und weil sie als akkurate Aufzeichnungen behandelt werden konnten, und damit als Begründungen für Politik, waren sie mit der Zeit willens jene Personen zu unterstützen, die Zahlen zur Erzeugung von Sinn einsetzten« (Poovey 1998, 97).<sup>civ</sup> Aus dieser Perspektive, so Poovey, ebnete die doppelte Buchhaltung der Cartesianischen Wissensproduktion einen weiteren Weg.

Es ist lohnenswert, vor diesem Hintergrund, die doppelte Buchhaltung und die sich wandelnde Verwendung der Tabelle als frühmoderne mediale Praktiken neu zu positionieren. Zeichnete sich die Moderne durch technische Reproduzierbarkeit aus, so sind in der Frühmoderne neben dem Buchdruck, welcher intensive medientheoretische Aufmerksamkeit erfahren hat, das Tabellieren und das Verfahren der doppelten Buchhaltung als kulturtechnische Praxen stärker in die Wahrnehmung zu rücken. Wir haben gesehen, wie die doppelte Buchhaltung neues Wissen in der Verknüpfung aus zeitlichen und sachlichen Logiken geschaffen hat, und wie dieses Wissen verwoben war mit der Emergenz der (damals neuen) Ökonomien des Frühkapitalismus. Wir haben außerdem gesehen, wie dieses ursprünglich kaufmännische buchhalterische Wissen seinen Eingang findet in die Wissensproduktion des Regierens. Mit diesen Genealogien lässt sich in einem nächsten Schritt die Bedeutung der Transaktion und ihre transformative Agency im Zuge der Globalisierung genauer beleuchten.

---

<sup>181</sup> Vgl. auch *Kalkül und Leidenschaft – Poetik des ökonomischen Menschen* (Vogl 2002, 223–246) und *Paper Machines* (Krajewski 2011).



## 2.3.4 Dispositive der Transaktionsverarbeitung

»Wissen sie, wenn sie über Bankgeschäfte nachdenken, dann hat sich nicht wirklich viel geändert. Automatisierte Clearing-Häuser arbeiten mehr oder weniger so wie früher. Die Ausrüstung ist moderner und billiger, aber ich vermute, dass viele Programme vor zwanzig oder dreißig Jahren geschrieben wurden« (Jim Gray 2002, 28).<sup>cv</sup>

Buchhaltung war eine der Motivationen für die Entwicklung von Datenbanken und wirtschaftlicher Anwendungssoftware. Die intensivierten buchhalterischen Prozesse des Industriezeitalters wurden von zunehmend komplexeren organisatorischen Strukturen begleitet: Beginnend mit Organisation von Büroräumen und Arbeitsabläufen (Meynen 2004; Krajewski 2011), der Taylorisierung von Kopfarbeit, maschinisiert in der Hollerithmaschine (Hartmann/Frank 2012; Grier 2003; Grier 2005), mechanischen Rechenmaschinen, elektronischen Rechenmaschine und dem Computer (Nake 1992; Ceruzzi 2003). Mit der Ausformulierung der Universalmaschine Computer als Medium verlagert sich der Entwicklungsschwerpunkt von der Hardware in den Bereich der Software – und softwarebasierte Datenbanksysteme sollen den Zugriff auf die Daten ermöglichen (Burkhardt 2015).

In der Medienhistoriographie der 1980er Jahre wurde die Emergenz von Technologien vor dem Hintergrund von Kriegserfahrung analysiert.<sup>182</sup> Vom erzählerischen Standpunkt her ist Krieg mit seinen immensen inneren Spannungen dankbarer als Fragen der Büroarbeit, Verwaltung, Logistik oder der sich massenhaft wiederholenden Berechnung. Dieser Argumentation entging, dass militärischen Operationen häufig Fragen der Logistik und der Verwaltung von Materialien und Informationen vorausgingen. Inzwischen kann die Besonderheit kriegerischer Auseinandersetzungen für die Entstehungsgeschichten von Computern anders produktiv gemacht werden mit dem Argument: Der Ausnahmezustand ›Krieg‹ erlaubte eine massive Verausgabung von Ressourcen, die staatlich gebündelt und mobilisiert wurden. Die Spezifik lag hier in der staatlichen Konzentration von Ressourcen, durch die sich eine Dimension der Logistik und des Verwaltens zieht.<sup>183</sup>

Spätestens ab Mitte des 19. Jahrhunderts ist ein erhöhter Bedarf zur Datenverarbeitung und Berechnung im Zuge der durch die industrielle Revolution dramatisch ausgeweiteten Warenproduktion feststellbar. Diese Form korrespondiert mit der Ausweitung naturwissenschaftlicher Wissensproduktion, die bereits seit dem 17. Jahrhundert ihren Lauf nahm. Großraumsäle mit Angestellten, die sich einer zu bearbeiteten ›Papierflut‹ ausgesetzt sahen, oder die zur mathematischen Berechnungen komplexer wiederkehrender Probleme herangezogen worden, sind Zeuginnen dieses Bedarfs. Es sei deshalb vorgeschlagen, die räumliche und prozessuale Anordnung dieser Großraumbüros als Vorbild für die im Entstehen begriffenen Datenverarbeitungsanlagen zu interpretieren, wie auch in den folgenden Abbildungen deutlich wird (vgl. Abb. 37 bis Abb. 39).

Zum Registrieren verfügten sie über Input und Output, d.h. eine räumliche Teilkonfiguration in der Daten eingegeben und ausgegeben wurden, beispielsweise auf Karteikarten aufgeschrieben oder,

---

<sup>182</sup> »Das distinktive Strukturmerkmal von [Friedrich] Kittlers Mediengeschichte [war] ihre enge Verbindung zur Militärgeschichte«, urteilt Hans Ulrich Gumbrecht als Herausgeber der Essays Kittlers (Gumbrecht 2013, 403). Vgl. auch die Bestandsaufnahme durch Claus Pias *Was waren Medien-Wissenschaften? Stichworte zu einer Standortbestimmung* (Pias 2011).

<sup>183</sup> SAGE, ein militärisches Projekt zur Luftraumüberwachung, das IBM für die US Air Force entwickelte und dessen Erfahrungen in das spätere zivile SABRE einfließen, kostete 20 Millionen Dollar für den Q7 Computer und 300 Mill. Dollar für jedes der 22 Leitzentren (Stand: 1962). Im Verhältnis dazu wurden die Anfangskosten 1959 für SABRE, das größte kommerzielle Informatisierungsprojekt seiner Zeit, auf 30 Mill. Dollar veranschlagt (Campbell-Kelly 2003, 44).

formalisierter noch, in Lochkarten gestanz. Als Output kann die Niederschrift der prozessierten Ergebnisse per Lochkarte, per Hand oder per Schreibmaschine gelten.

Eine weitere räumliche Organisationseinheit war jene Gruppe von Mitarbeiterinnen, welche die Daten berechnete oder prozessierte, bis in das 19. Jahrhundert im Kopf und per Hand, danach außerdem unter Zuhilfenahme entsprechender mechanischer und elektromechanischer Apparate. Im elektronischen Computer verwirklichte sich dieser Ort im Begriff der CPU, der Central Processing Unit.<sup>184</sup> Neben der Rechenleistung der menschlichen Computer, wurden Prozesse des Sortierens, Summierens und Gruppierens von Informationen, ebenfalls räumlich getrennt, per Hand oder mithilfe entsprechender Maschinen (z.B. Hollerith-Tabulatoren) durchgeführt. Hier sind auch bereits Praktiken des Delegierens zu identifizieren, wenn das Prozessieren bestimmter Informationen als eine Operationskette aus Adress- und Speicheroptionen zwischen maschinell-mechanischen und humanen Akteuren verläuft.

Davon unterschieden ist der Ort des Speicherns und der Ablage. Die Schrankmeter mit Karteikartensäfeln, also die Übernahme bibliografischer Verzeichnungsweisen in das Feld der ökonomischen Bürokratie, sind ihre Funktion nach prototypisch bereits Dateisysteme/Datenbanken.

Der Übergang zur elektronischen Rechenmaschine kennzeichnet sowohl eine Kontinuität als auch einen Umbruch. In Karteikarten, Tabellen und Formularen, und später der Organisation der Rechenmaschinen, der Speicher und Prozessoren sind die Daten, mit Markus Krajewski gesprochen, in Form gebracht – ein weiteres Indiz der Verräumlichung zur Informationsverarbeitung. Ein Bruch entsteht durch die Maschinisierung ehemals menschlich verausgabter Arbeitskraft, der zur Abwesenheit einer bestimmten, häufig weiblich dominierten, Schicht von Wissensarbeiterinnen führte. Mit diesem Verständnis soll eine Historiographie entworfen werden, die nicht auf die medientechnische ›Revolution‹ abzielt, sondern auf Kontinuitäten und Verschiebungen, die sich aus den Alltagspraktiken ergeben.

Viele der maschinell ersetzten Tätigkeiten waren bereits der Taylorisierung, d.h. ihrer Aufspaltung in kleinste Einheiten unterworfen worden. Ein Programmierer aus dem VISA-Karten Projekt der 1970er berichtet von seiner Begegnung mit einer Angestellten, deren Tätigkeit er auf die Informatisierbarkeit hin untersuchen sollte: »Es war beschämend, einmal, als ich in unsere Premium-Collection Abteilung ging und zu einem Mädchen [orig.: *girl*, gemeint ist die Angestellte F.H.] dort und sagte ›Also, was ist Deine Aufgabe hier?‹ und sie sagte, ›Ich erledige die CP11er‹, und ich sagte ›Also, was ist ein CP11er?‹ und sie sagte ›Naja, es ist eine Transaktion. Ich übertrage diese Nummer hier in diesem Kasten auf dieses Blatt Papier‹. Und ich sagte ›Aha, und warum machst Du das?‹. Sie wusste es nicht. Nicht ein Schimmer. Weil sie es nicht zu wissen brauchte. Ihre Aufgabe war zu einem Punkt reduziert worden, an dem sie tatsächlich nichts wissen musste. Wohlgermerkt, ich glaube nicht, dass sie den Job deswegen mehr mochte. Und ich denke nicht, dass sie ihn durch ihr Nicht-Wissen besser erledigte« (Fincham (Hrsg.) 1994, 205).<sup>cvii</sup> Die maximale Entfremdung der Arbeitenden von den durch sie durchgeführten Handlungen erweist sich als doppelschneidig: Einerseits erscheinen sie schlichtweg als inhumane Tätigkeiten über deren maschinellen Ersatz man froh sein konnte, andererseits war genau diese tayloristische Aufspaltung in kleinste Aktionen erst Voraussetzung für deren maschinelle Übersetzung.

---

<sup>184</sup> Central Processing Unit ist die amerikanische Variante, welche in der ›Unit‹ auch eine Zusammenfassung, eine räumlich-funktionale Einheit impliziert. Konrad Zuse nannte diese räumlich-funktionale Einheit das ›Rechenwerk‹, da er sich mit wiederholenden kleinteiligen aerodynamischen Berechnungen konfrontiert sah, die er in einem ›Werk‹, einem Ort der industriellen Produktion, organisiert, formalisiert und automatisiert.



Abb. 36: Wissens-Arbeiterinnen an den Karteikartenschränken des Universellen Bibliographischen Verzeichnis des Mundaneums, ca. 1900. Transaktionsdaten treten hier vor allem auf, wenn kostenpflichtige Benutzeranfragen verbucht und in Rechnung gestellt werden. Anhand von Transaktionsdaten ist nachzuvollziehen, ob eine spezifische Anfrage beantwortet war und der damit verbundene Gesamtvorgang des Suchens, der Korrespondenz, der Rechnungsstellung und Bezahlung und der Versendung des gefundenen Wissensobjektes als abgeschlossen gelten konnte (Mundaneum, Archive Centre of the French Community of Wallonia-Brussels and Temporary Exhibition Space).



Abb. 37: Arbeitssaal der Computer Tabulating Company, Washington 1905. Die Frauen im Vordergrund übertragen mithilfe des IBM Keypunch statistische Daten auf Lochkarten. Hinter der Tischreihe stehen die Supervisorinnen, die den Ablauf überwachen. Hinten links im Bild sieht man Sortiermaschinen, sowie Tabulatoren welche als Central Processing Unit das Summieren der Daten ermöglichen. Im Hintergrund rechts sieht man Aktenschränke, die auf das spätere Dateisystem, beziehungsweise die Datenbank verweisen (Harris & Ewing Collection, Library of Congress, LC-H25- 91219-KG [P&P]).



Abb. 38: Büro einer Versicherungsfirma mit Women of Color als Büroarbeiterinnen, Chicago, April 1941. Auf den Schreibtischen stehen Rechenmaschinen und Schreibmaschinen, im Hintergrund rechts sind Karteikartenschränke zu erkennen. Nachdem während der wirtschaftlichen Depression der 1930er Jahre zuerst WoC entlassen wurden, kamen sie als »Reservearmee« in der Zeit des zweiten Weltkrieges als Büroarbeiterinnen zum Einsatz (Lee, Russell/Library of Congress, Prints & Photographs Division, FSA/OWI Collection, LC-DIG-fsa-8a29933).



Abb. 39: Dateneingabe bei der National Girobank, Bootle/ Lancashire ca. 1969. Mitarbeiterinnen an Terminals für Schecks und Überweisungsscheine, die nicht automatisch erfassbar waren. Es wurden überwiegend Frauen eingestellt, da diesen geringere Löhne gezahlt werden konnten. Die bis zu 142 Datenerfasserinnen waren in Gruppen von 12–15 dem Supervisor unterstellt, welcher Informationen aus einem Teleprinter bezog. Wurden für den Computer fehlerhafte Daten eingegeben, die der Logik des Datenmodells widersprachen, wurde das entsprechende Keyboard gesperrt, um die erneute Dateneingabe zu initiieren (Standard Telephones and Cables Ltd. In: (Dummer/Thomson/Robertson 1971, 919ff.)).

### 2.3.5 Ein Ticketverkäufer wird zum Informationsanbieter

Als nächstes soll ein frühes – und damals das größte – Projekt der logistischen In-Formatisierung diskutiert werden, welches privatwirtschaftlich in Kooperation zwischen den Firmen International Business Machines und American Airlines entstand: SABRE (Semi-Automatic Business Research Environment). Flugbuchungssoftware in den USA sind eine der ersten vernetzten und lokal verteilten Datenverarbeitungsfigurationen auf Computerbasis mit grundlegenden Implikationen. Im Laufe dieser Entwicklung wandelten sich diese Figurationen von beigeordneten Arbeitshilfen zu Grundlagen des Geschäftsbetriebes, die ebenso wichtig wurden, wie die Flotte der Flugzeuge selbst (Copeland/Duncan G./McKenney 1988, 353). SABRE kann auch als paradigmatisch für eine Flexibilisierung von Transaktionen gelesen werden, die erst mit programmgesteuerten, vernetzten, das heißt distanten Informationstechnologien überhaupt möglich ist, und in der sich eine Kongruenz von Globalisierung (weltweit gesicherter Transaktion) und Entwicklung der Informationssysteme nicht nur unterstellen, sondern nachweisen lässt. Wie sich dieser analytische Anteil an der Datenprozessierung zunehmend in Geschäftsprozesse einschreibt, soll in den folgenden Abschnitten dargestellt werden, um herauszuarbeiten, welcher epistemischer Mehrwert dem Verarbeiten von Transaktionen in Datenbanken inne wohnt. Den in den 1930ern etablierten, über Telefonleitungen und Telex zentral vernetzten, manuellen Buchungsvorgang beschreiben die Autoren Duncan Copeland und James McKenney in ihrer detaillierten Studie *Airline Reservations Systems: Lessons From History*. »Die manuellen Systeme, die damals eingesetzt wurden, hielten Inventare der Sitzplätze für einen bestimmten Flug zum Zeitpunkt des Abfluges vor. Die für Reservierungen zuständigen Mitarbeiter konnten Plätze eines Fluges buchen, nachdem sie deren Verfügbarkeit anhand großer Übersichtstafeln überprüft hatten, die sich in jedem Reservierungsbüro befanden. Nachdem ein Platz verkauft wurde, sendete ein Mitarbeiter per Telefon oder Telegraf eine Buchungsnachricht an das Reservierungsbüro, das sich am Abflugort befand. Beim Empfang einer Buchungsnachricht, verringerte ein Angestellter die Anzahl der verfügbaren Sitze aus dem Inventar, welches in einem Loseblatt-Ordner vorgehalten wurde. Wenn die Zahl der verfügbaren Plätze für einen Flug unter eine spezifische Schwelle fiel, wurde eine ›Verkaufsstopp‹-Nachricht an alle Reservierungsbüros gesendet, die auf die Übersichtstafeln übertragen wurde« (Copeland/Duncan G./McKenney 1988, 354).<sup>cvi</sup> Das Reisebüro erfragte bei den Passagieren deren Passagierdaten, sobald ein Flug erfolgreich gebucht war. Sie wurden in Formularen notiert und per Telefon oder Fernschreiber an das Büro der Fluglinie am Abflugort übermittelt und dort erneut in einem Formular niedergelegt. Diese gesammelten Passenger-Name-Records (PNR) glich das Büro kurz vor Abflug erneut mit der Liste der verfügbaren Sitze ab, ein Prozess, in dessen Zuge häufig und nahezu zwangsläufig inkonsistente Daten entstanden. Da die Überprüfung erst kurz vor Abflug erfolgen konnte, war es notwendig, in den Flugzeugen genügend Restsitze vorzuhalten – eine beständige Abwägung zwischen Servicequalität und Flugzeugkapazität (Abb. 40).<sup>185</sup>

Ausgangspunkt für elektronische Buchungsfigurationen war dieser bereits etablierte manuelle Buchungsvorgang: Zuerst war die Verfügbarkeit eines Fluges zu überprüfen, dann verfügbare Sitze entsprechend der Buchungen oder Absagen zu aktualisieren und abschließend erfolgte die Übertragung

---

<sup>185</sup> Die Buchungen wurden seit Mitte der 40er Jahre mit einem elektromechanischen System namens Revisor unterstützt, welches ähnlich einem Telegraph Informationen übermitteln konnte und von einer Firma namens Teleregister hergestellt wurde. Teleregister wiederum war auf die Hardware zur Übermittlung von Börsennachrichten konzentriert und als ursprüngliche Ausgründung aus Western Union Teil jener technischen Infrastruktur, welche Bank-Transaktionen im Zuge der Telegraphie in den Vereinigten Staaten etabliert hatte. Die Suche nach einem medientechnischen Vorläufer der transaktionsorientierten Buchungssysteme der 1950er bleibt also nicht ohne Verweis auf die Ende des 19. Jahrhunderts etablierte Fernmeldetechnik der elektrischen Telegraphie.

der Passagier-Stammdaten. Die Flugbuchung war ein Prozess, der einen konkurrierenden Zugriff auf diese Daten von geographisch entfernten Orten benötigte. Das Informationstechnische a-priori von Fluglinien der späten 1950er Jahre bestand in dem Problem, einerseits ein Inventar an verfügbaren Sitzplätzen aktuell zu halten und andererseits die vergebenen Sitzplätze mit den Personendaten zu verknüpfen, so dass zur Abflugzeit ein aktuelles Register der zustiegsberechtigten Personen vorlag. Fehlbuchungen traten unter den Bedingungen der telegrafischen oder telefonischen Übermittlung häufig auf. Ein Fehler aller 12 Buchungen war Normalität (Campbell-Kelly 2003). Sie wurden neben dem Ärger der Passagiere zum Hindernis für eine Ausweitung des Flugverkehrs insgesamt. Dies bedeutet umgekehrt, dass die Automatisierung der Transaktionsverarbeitung einen deutlichen Anteil an der Globalisierung logistischer Prozesse hatte. Die Anschlussmöglichkeiten an Flüge anderer Anbieter waren kompliziert zu buchen, da jene ebenfalls Schwierigkeiten hatten, ihre Kapazitäten zeitnah darzustellen. Als American Airlines mit IBM über die Entwicklung der elektronischen Buchungsfiguration SABRE übereinkam, erhoffte sich das Unternehmen eine Software, welche Passagiere und Sitzplätze einander zuordnen kann, die Kommunikationsgeschwindigkeit zwischen den Fluglinien optimiert, die Sitzplatzkapazitäten aller Anbieter transparent macht, um Anschlüsse zu garantieren, die Flugrouten der Passagiere ausdrucken kann, entsprechende Boardingpässe ausstellt und schließlich Zugriff über vernetzte Terminals für lokale Reiseagenturen ermöglicht.<sup>186</sup>

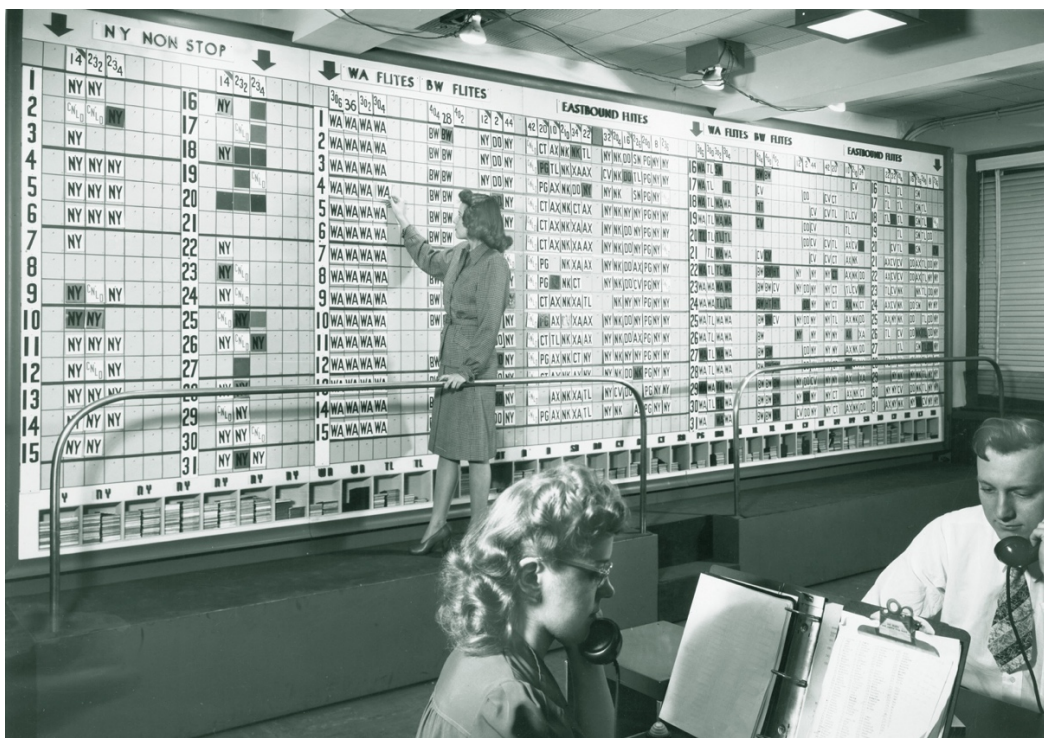


Abb. 40: United Airlines Reservierungstafel, ca. 1930er Jahre (Smithsonian National Air and Space Museum Washington, 2000-9721)

Die Ausgangs-Konfiguration von SABRE bestand aus zwei IBM 7090 Computern, die der Ausfallsicherheit dienen und einander spiegelten, mit 16 IBM 1301 Festplatten und sechs zusätzlichen IBM Trommelspeichern. Mittels 16 Telefonleitungen sollten 1100 Terminals verbunden werden, mit einer

<sup>186</sup> Detailliert wird in der Studie *Airline Reservations Systems – Lessons From History* (Copeland/Duncan G./McKenney 1988) nicht nur die Situation von American Airlines, sondern auch der Konkurrenten United Airlines, TWA, Eastern und Delta und deren Erfolge und Scheitern informatorischer Projekte geschildert.

mittleren Antwortzeit von 3 Sekunden (statt der durchschnittlich 45 Minuten, die eine komplette manuelle Buchung bis dahin in Anspruch nahm), wobei 26.000–40.000 Passagierdateien (Passenger Name Records) täglich bearbeitet werden konnten. Als erster Ort sollte im April 1963 Hartford, Connecticut mit dem 150 km entfernten AA-Rechenzentrum in Briarcliff, NY verbunden werden. Eines der Probleme in der Planung war es, nicht nur die übliche Anzahl von Transaktionen abwickeln zu können, sondern auch Höchstlasten zu puffern. Es ging darum, bereits von vornherein Skalierbarkeit in die Software-Figuration einzubauen, da eine Erweiterbarkeit mit Speicher und Telefonleitungen nicht innerhalb von Tagen, sondern nur innerhalb von Monaten erreichbar war. In einem nicht-computerisierten Ablauf konnten Überstunden angeordnet werden, um eine Transaktionsspitze abzufangen, doch eine Kapazitätsausweitung war bei Umstellung auf Maschinenteknik nur möglich, wenn die technischen Speichermittel dafür zur Verfügung standen. Aus der Perspektive ihrer Maschinisierung tritt die Transaktion somit als ein Input-Output-Problem auf, denn nicht die Rechenkapazitäten schränkten die Anzahl der maschinisierbaren Vorgänge ein, sondern die Frage, wie diese Vorgänge dem Maschinischen überhaupt zugeführt werden können.

Anhand der bei American Airlines betriebsintern vorliegenden Auslastungszahlen für tägliche Anrufe sollten die Kapazitäten geplant werden. Auf Nachfrage stellte sich heraus, dass die intern verwendeten Angaben keine tatsächlichen statistisch erhobenen Daten waren, sondern aus Jahresvorgaben des Managements bestanden. Für den zuständigen Projekt-Manager bei American Airlines, Fred Plugge, war dies Anlass, eine neue Abteilung zu gründen, die sich System Measurements nannte und die die benötigten Transaktionsanalysdaten erarbeitete. Es erwuchs ein Projekt, indem die Anzahl der bearbeiteten Passenger Name Records täglich ermittelt und mithilfe von zu verfeinernden Mitteln die zukünftige Auslastung modelliert wurde – und eine Sicht auf Daten, die deren epistemischen Mehrwert in den Blick rückte (DeSofi 2009, 91f., 100f.; Copeland/D.G./Mason/McKenney 1995, 37f.).

Wie sah diese In-Wert-Setzung von Transaktionsanalysen im Detail aus und wie wurde sie in die sich ändernden Geschäftsabläufe eingebettet? Neben den erwarteten Produktivitätssteigerungen, zum Beispiel mittels Ersetzung von Menschen durch Maschinen (Copeland/Duncan G./McKenney 1988, 366), höherer Datenübertragungsrate oder der besseren Flugauslastung, zog American Airlines ab Mitte der 1960er eine Reihe unerwarteter Gewinne aus ihrem Investment. Der Blick soll an dieser Stelle konzentriert werden auf die Informationsrendite aus Transaktionsanalysen. American Airlines begann in der Systems Measurements Abteilung, die Reservierungsdaten auszuwerten, um ein Fine-Tuning der Betriebsabläufe und der Marketingstrategien vorzunehmen. »Diese akkuraten Passagierlisten eröffneten die Möglichkeit, Unter- und Überbuchungen zu steuern, um Passagierservice und Auslastungskapazität (der prozentualen Nutzung vorhandener Sitzplätze) gleichermaßen auszutarieren. Mithilfe von Passagierdatenbanken konnten alternative Entscheidungsszenarien modelliert werden« (Copeland/Duncan G./McKenney 1988, 357).<sup>cvi</sup>

Das ursprüngliche Buchungssystem wurde bald um Reservierungen erweitert, womit Kostenvorschläge, vorzeitiger Check-In, die Ausgabe von Boarding Pässen und die Erstellung potenzieller Reiserouten möglich wurden. Diese Reservierungsfunktion versetzte Reisebüros in die Lage zu deutlich vereinfachten Preisvergleichen, womit sich die Marktsituation änderte: Der Fokus verlagert sich von der bloßen Bereitstellung von Kapazitäten auf diejenigen, der die Dienstleistung am günstigsten zur Verfügung stellen konnte.

Im Zuge dieser Entwicklung wird ein diagrammatisches Detail relevant: Welche Flüge stehen auf den Bildschirmterminals der Buchungsagenten in der Liste oben? Vereinfacht gesagt, waren es die Flüge derjenigen, die die entsprechenden Buchungssysteme anboten: American Airlines im Falle von SABRE (Abb. 41) und United Airlines im Buchungssystem APOLLO. Studien aus jener Zeit legen nahe, dass

bis zu 70% aller Buchungen über die erste von mehreren Bildschirmseiten des Terminals erfolgten, und dass mit dieser privilegierten Darstellung das Marktgeschehen ›verzerrt‹ würde. Die Betreiber von SABRE und APOLLO verfügten über Einblick in die Transaktionsdaten und zwar nicht allein die eigenen, sondern auch jene der Konkurrenz. Die daraus gewonnenen Analysedaten ermöglichten ein neue Qualität der Preisgestaltung und die Anzahl verschiedener Preise wuchs innerhalb kürzester Zeit stark an.



Abb. 41: SABRE Buchung. Das spezielle Interface erlaubt über codierte Papierkarten die Auswahl aus einer Liste zur Verfügung stehender Flüge. Über einen Button konnte ein spezifischer Flug im Hauptrechenzentrum per Telefonleitung angefragt werden und stand dieser zur Verfügung, leuchtete eine entsprechende Lampe auf. Die Buchung konnte durch die Mitarbeiterin bestätigt werden und die Passagierdaten anschließend per Tastatur eingegeben werden. Ein Set kostete 16.000 US Dollar (entspricht derzeit ca. 137.000 US Dollar). Ab 1976 wurde dieses opto-elektrische Interface durch Bildschirmterminals ersetzt (IBM Archive, 1962).

Die Häufigkeit der Preisänderungen veränderte sich »von halbjährlich, zu monatlich, zu halbmonatlich, und in einigen Fällen täglich« (ebd. 362).<sup>cix</sup> Programme zur Kundenbindung, die auf den tagesaktuellen, auf ein Individuum zu beziehenden Transaktionsanalysen basieren, werden einerseits technisch möglich und folgten andererseits den Bedürfnissen einer Branche, die einen Service anbietet und keine Ware. Mit den sukzessiven Versionen von SABRE gelang es IBM ein neues Paradigma der Informationsverarbeitung in eine Abfolge ineinandergreifender Mensch-Maschine-Interaktionen zu etablieren: On-Line Transaction Processing (OLTP) versprach die ›unmittelbare‹ Verarbeitung von Transaktionsdaten als ›Überwindung‹ der zeitverzögerten Stapelverarbeitung. »American Airlines perfektionierte das Management komplexer Preisstrukturen und profitierte nachhaltig durch den Einsatz von Wissen, das es aus SABRE zog« (ebd. 368),<sup>cx</sup> so die Zusammenfassung von Copeland und McKenney. Wir sehen hier beispielhaft, wie sich die Transaktion doppelt vollzieht: Einerseits als menschliche Praxis und andererseits als deren Abstraktion und Nachvollzug im Computer. »Die Anwendungsprogramme selbst waren ›wieder-eintrittsfähig‹ aufgebaut, was bedeutete, dass jede Routine ihren eigenen Rechenvorgang aufräumte, selbst wenn die Passagierbuchung noch nicht vollständig war. Mit diesem Multiprogramming-Ansatz konnten mehrere Reservierungsanfragen gleichzeitig abgearbeitet werden, die solche üblichen Abläufe wie das Überprüfen der Verfügbarkeit,



die Eingabe von Passagierdaten oder die Bestellung spezieller Mahlzeiten gemeinsam hatten. Individuelle PNR-Daten wurden in einem ›Agenten-Sammelgebiet‹ vorgehalten und nur bei Aktivierung des Transaktions-Endprogrammes wurden all die Daten einer bestimmten PNR-Datei dauerhaft abgespeichert« (Head 2002, 34).<sup>cxi</sup> Die computerisierte Transaktion erscheint als Nachvollzug, als Aufzeichnung menschlicher Aktion und kann erst als abgeschlossen gelten, wenn auch die zugrundeliegende menschliche Aktion abgeschlossen ist. Gleichzeitig bestätigt sich in dieser Beschreibung der doppelte Charakter des Transaktionsbegriffes: Einerseits als ökonomisch relevante Aktion (Flugbuchungen) und andererseits in der Interaktion mit einem informatorischen Prozess, der den Anfang und das Ende einer Aktion klar abgrenzbar festschreibt.

### 2.3.6 On-Line Transaction Processing (OLTP)

»Die Worte ›Transaktion‹ und ›Aufgabe‹ haben in CICS die gleiche Konnotation und werden innerhalb der Publikation austauschbar eingesetzt« (IBM 1972).<sup>cxii</sup>

Am Beispiel von American Airlines konnten bereits die Auswirkungen diskutiert werden, die eine Umstellung auf Online-Transaktionsverarbeitung und die Auswertung von Transaktionsdaten nach sich zogen.

Im Folgenden soll in einem Durchgang durch die frühen Softwareprodukte und Hersteller elektronischer Datenbanken eine Reihe wichtiger Übergänge, welche zum heutigen Begriff der Transaktion in elektronischen Datenbanken geführt haben, dargestellt werden.

Aus Sicht der Informatik besteht eine (manuelle) Transaktion nicht aus einer einzelnen Aktion, sondern aus einer idealtypischen Abfolge von Teilaktionen, die eine gesamte Transaktion ergeben. Wenn beispielsweise eine Bankkundin eine Summe von ihrem Sparkonto auf das Girokonto überweisen möchte, so ist der erste Teilschritt eine Überprüfung, ob das Sparkonto über mindestens diesen Betrag gedeckt ist. Ist dies der Fall, kann der Betrag im zweiten Schritt vom Sparkonto abgezogen werden. In einem dritten Schritt kann der Betrag dem Girokonto gutgeschrieben werden. Alle drei Teilaktionen hängen unmittelbar zusammen. Sollte beispielsweise der letzte Schritt aus technischen Gründen (z.B. durch einen Netzwerkfehler) scheitern, würde die Gutschrift auf das Girokonto fehlen, jedoch der Abzug vom Sparkonto bereits geschehen sein und die Kundin ginge des Betrages verlustig. Es musste ein Rahmen geschaffen werden, in dem entweder alle drei Teilschritte erfolgreich durchgeführt werden können, oder bei einem Scheitern eines Teilschrittes, die Inhalte der Datenbank auf den Anfangszustand zurück gesetzt werden konnten.

#### 2.3.6.1 Nicht-Linearer Speicherzugriff

Im Alltag der Büro- und Verwaltungsarbeit erfolgen nicht durchgehend immer gleiche Transaktionen. Verschiedene, sich überschneidende Arten der Transaktion mit jeweils distinkten Einzelschritten sind erforderlich. Solang dies manuell vorgenommen wurde, war es insofern kein Problem, als dass Angestellte auf verschiedene Speicherorte der zugehörigen formatierte Information, z.B. in Karteikästen, in zufälliger Weise zugreifen konnten und Absprachen treffen konnten, um Überschneidungen zu lösen. Mit der Maschinisierung der Vorgänge wurde der Zugriff auf verschiedenste Informationsobjekte schwieriger, da diese entlang einer linearen Maschinenlogik des Daten-Eingangs und -Ausgangs, z.B.

des Lochkartenstapels oder des Magnetbandes organisiert waren. Das Aufeinanderfolgen des einen Informationsobjektes auf das nächste Informationsobjekt deflexibilisierte die Verarbeitung verschiedener aufeinanderfolgender Transaktionen und förderte eine einschränkende Reorganisation von Büroabläufen entlang von Transaktionen, die auf die immer gleichen Datenobjekte zugriffen. In der Alltagspraxis erwies sich dies als Hemmschuh für eine weitere Ausweitung von Transaktionsdatenverarbeitung. Ein Jahr, bevor der weltweit erste Satellit ins All startete, 1956, stellte IBM mit dem IBM RAMAC, welcher erstmals Festplattentechnologie mit *Random Access*, also nicht-linearem Speicherzugriff ermöglichte, einen Lösungsansatz für dieses Problem vor. IBM erklärte in der Produktbroschüre von 1957 anhand eines Beispiels anschaulich, wie der Engpass der Ein- und Ausgabe von Transaktionsdaten mit Hilfe des sogenannten Inline-Processings bewältigt werden sollte. Dargestellt wurde der Vorgang anhand von Angestellten, die Transaktionen über eine Reihe verschiedenster Konten vornehmen, zum Beispiel für Bargeld, für Rohmaterialien, für verarbeitete Materialien, für Inventar und für Einkünfte aus Verkäufen.

»Da der Angestellte direkten Zugang [im engl. Orig. ›direct access‹] zu all diesen Konten hat, kann er jede Transaktionen vollständig abwickeln, bevor er mit der Erledigung der nächsten beginnt. Diese Buchungsmethode wird als ›Inline-Processing‹ bezeichnet. Dieses war in der Vergangenheit in automatisierten Buchungssystemen unpraktisch gewesen, aufgrund der Schwierigkeit, einzelne Datensätze innerhalb großer Dateien zu erreichen und zu verändern« (IBM 1957, 5).<sup>cxiii</sup> Die auf den Festplatten des IBM RAMAC Systems erreichte Verräumlichung des Informationszugriffes unterscheidet sich signifikant von der Linearität von Lochkartenstapeln und Magnetbändern und ermöglicht eine Verdichtung der Transaktionsverarbeitung innerhalb des zur Verfügung stehenden Acht-Stunden-Arbeitstages auf ca. 10.000 Transaktionen, also auf rund 20 Transaktionen pro Minute (ebd., 10). IBM stellte die Entwicklung der Random-Access Festplatte nicht in den Zusammenhang wissenschaftlicher Berechnung oder militärischer Echtzeitsysteme, sondern in das Kerngebiet der Firma, die Verarbeitung ökonomischer Daten. »Die IBM RAMAC hat als Zentrum ein Speichergerät mit nicht-linearem Speicherzugriff, welches die Ablage von fünf Millionen Zeichen an Wirtschaftsfakten in der Maschine erlaubt« (ebd., 5).<sup>cxiv</sup> Aus dieser spezifischen Perspektive kann die technologische Entwicklung der Festplatte als ein Ergebnis der Auseinandersetzung mit buchhalterischen Bedürfnissen – und spezifisch mit dem Bedarf, Transaktionen abzubilden und vorzunehmen – beschrieben werden.

#### 2.3.6.2 Priorisieren, koordinieren und parallel verarbeiten

Festplattentechnologie wurde zur unmittelbaren technischen Voraussetzung für die Entwicklung der ersten elektronischen Datenbanken im heutigen Sinne. Abweichend von der durch Thomas Haigh entwickelten Lesweise als Management-Informationssystem (Haigh 2001), wird in der vorliegenden Studie diese Entwicklungen als Figuration zur Transaktionsverarbeitung diskutiert, um herauszuarbeiten, welche grundlegende Bedeutung der Transaktion als Motiv für die Entwicklung elektronischer Datenbanken zukommt.

Im Interview mit Haigh bezeichnete der Ingenieur Charles Bachman die Einführung von Festplattentechnologie und die damit einhergehenden softwaretechnischen Neuerungen als »grundlegende Verschiebung« (Bachman 2004, 53). Während vorhergehende listenorientierte Programme wie der Report-Generator 9PAC für die Abarbeitung von Stapeln und Magnetbändern konzipiert war, hing der von Charles Bachman, Stan Williams and Bill Helgeson bei General Electric entwickelte Integrated Data Store (IDS) vom nicht-linearem Speicherzugriff ab. Dem Modell IBM RAMAC und der dort etablierten Speicheradressierung stand er jedoch skeptisch gegenüber: »Die [automatische] Berechnung von

Adressen erwies sich auf dem RAMAC als unpraktisch. In der Folge war der RAMAC nicht sehr erfolgreich« (ebd. 52)<sup>cxv</sup> – einer der Gründe, warum General Electric auf Festplatten der Firma Data Products setzte.

Der Integrated Data Store, der heute als früheste elektronische Datenbank nach dem Netzwerkmodell angesehen wird und später als eigenes Produkt weiter entwickelt wurde, war zum Zeitpunkt seiner Entwicklung ein »Nebenprodukt« (ebd. 45) des Informationsverarbeitungs-Systems MIACS.<sup>187 cxvi</sup> Zu MIACS und dem Integrated Datastore gesellte sich eine weitere Softwareinnovation, die als *Problem Controller*<sup>188</sup> bezeichnet wurde, ein Verteiler für abzuarbeitende Transaktionen. Dieser Transaktionsmonitor erlaubte es, den zu verarbeitenden Transaktionen verschiedene Prioritätsstufen zuzuweisen, sodass Transaktionen höherer Priorität zuerst abgearbeitet wurden.

Während also in der Stapelverarbeitung alle Transaktionen nach Reihenfolge des Eingangs über Nacht abgearbeitet wurden, ermöglichte die Priorisierung mittels Problem Controller eine unmittelbare Verarbeitung von Transaktionen. Sobald eine Transaktion anliegt, wird sie verarbeitet. Falls keine Rechenkapazitäten zur Verfügung stehen, wird sie in eine Warteschlange verschoben, bis Transaktionen höherer Priorität abgearbeitet waren und die Ressourcen für niedrigere Prioritäten freigegeben wurden (ebd. 81). Softwaretechnisch wurde somit eine Verdichtung der Transaktionsleistung organisiert, die die Rechenkapazitäten der Maschine intensiver ausnutzte als zuvor und Stillstandzeiten verringerte.

Eines der Beispiele, die Bachman für den Einsatz von Integrated Datastore und Problem Controller zur Transaktionsverarbeitung erörterte, war das Comprehensive Operating System (WEYCOS 1), welches die Produktionsketten der holzverarbeitenden Firma Weyerhaeuser Lumber abbilden sollte. Bis Mitte der 1960er Jahre wurde in der Firma manuell gebucht. Die Verkaufsbüros verfügten über Fernschreiber mit Lochband. Die jeweiligen Versanddepots (unterschieden nach geographischer Region und Produktangebot) wurden per Fernschreiber über erfolgte Verkäufe von Holz informiert. Jedes Versanddepot disponierte, abhängig von den eingehenden Bestellungen, seine Inventarlisten und versendete die Waren. Ebenfalls per Fernschreiber informierten die Mitarbeiter die Verkaufsbüros über den erfolgten Versand.

Mit Einführung der neuen Software erfolgte der Buchungsvorgang in Tacoma, zentralisiert über einen GE 235-Computer mit WEYCOS 1, Integrated Datastore und Problem Controller. Ermöglicht wurde dies durch einen vorgeschalteten General Electric Datamet 30-Computer, welcher mit bis zu 100 Fernschreibern verbunden war, die zur Eingabe und Ausgabe dienten. Damit zählte WEYCOS 1 neben SABRE zu den frühesten elektronischen Online-Transaction-Processing-Figurationen weltweit, die nicht mehr im Modus der Stapelverarbeitung sondern des direkten Zugriffes auf Rechenressourcen operierten. »Diese Fernschreiber waren über die gesamten Vereinigten Staaten verteilt, in allen Weyerhaeuser Verkaufsbüros, Sägewerken und Lagerhäusern. Sozusagen war diese Version des Problem Controller ein tatsächlich Nachrichten-orientiertes Betriebssystem. Einkommende Nachrichten konnten an Dritte gesendet werden, oder sie kamen herein und wurden an den GE 235-Computer übermittelt, wo sie per Problem Controller verteilt wurden« (ebd. 83).<sup>cxvii</sup> Ein Beispiel für eine höhere Prioritätszuordnung war das Anlegen neuer Kunden in der Datenbank, denn erst wenn diese angelegt waren, konnten ihre Bestellungen überhaupt verarbeitet werden.

---

<sup>187</sup> MIACS, so berichtete Bachman, stand für Materials Information And Control System, für Manufacturing Information And Control System oder für Management Information And Control System, abhängig davon aus welcher Perspektive die Beteiligten das Acronym verwendeten (Bachman 2004, 45f.).

<sup>188</sup> Der Produktname Problem Controller geht zurück auf die Selbstpositionierung von General Electric als »Problem Solvers«. Jede Transaktion wurde durch Bachman und seine Kollegen als »Problem« gesehen, welches es unter Kontrolle zu bringen gelte (Bachman 2004, 46).

Ermöglicht wurde das durch einen Wandel in der Datenverarbeitung: Während bis dahin die Informationsobjekte vom Rechenprozess aus gedacht wurden und in den Rechenprozess eingeschleust, verarbeitet und wieder ausgegeben wurden, waren im Integrated Data Store die Informationsobjekte »bereits vor Ort«, und es war das Programm, welches durch den Adressbereich der Datenbank, mittels sogenannter Pointer, navigierte (ebd. 55). Die elektronische Datenbank wurde zum Zentrum im Zentrum: sie diente im zentralisierten Mainframe-Computer als zentraler Ablageort für jegliche Daten. Informationsobjekte mussten nicht länger aus verschiedensten Kontexten »geborgen« werden, sondern waren nun dauerhaft abrufbereit gelagert.

Listengeneratoren, wie der 1959 von IBM-Nutzern entwickelte 9PAC, waren in der Lage, eine Datei, in der jede Zeile einen Datensatz darstellte, von oben nach unten auszulesen und daraus komplexe Berichte zu generieren, in denen die Daten in eine andere Sortierreihenfolge gebracht, gruppiert und summiert wurden. Die Zusammenführung verknüpfter Daten, z.B. wenn eine Autorin mehrere Bücher geschrieben hatte, aber diese Bücher auch über Co-Autorinnen verfügten, konnten nur über komplizierte, verschachtelte Schleifen-Strukturen programmiert werden.

Bachman, der an der Entwicklung von 9PAC 1957 teilweise beteiligt war, verfolgte im Integrated Datastore eine weiter entwickelten Ansatz, der auf eine netzwerkartige Verknüpfung von Datensätzen abzielte. Das vereinfachte auch die Bearbeitung von Transaktionen, die selten für sich allein in einer Datei abbildbar waren und häufig auf verschiedenste Datenkörper zugreifen mussten. Die Eltern-Kind-Hierarchie von Datenknoten ermöglichten es, eine Transaktion, d.h. eine Abfolge von Aktionen und Daten, die diese Aktionen kodifizieren, als Kind-Elemente eines Elternknotens zu behandeln.

Card Number	Input		Field Name	Control Break	Accumulate	Output	
	Record Type	Field Number				Line Number	Rightmost Print Position
1	01	0003	PROJECT NUMBER	1			
2	01	0006	ESTIMATED HOURS			T10	037
3	01	0007	PROJECT NAME			T10	009
4	02	0006	SUB-PROJECT NUMBER			D06	007
5	02	0007	CURRENT HOURS			D06	023
6	02	0007	CURRENT HOURS		X	T10	023
7	02	0010	TOTAL HOURS			D06	030
8	02	0010	TOTAL HOURS		X	T10	030

Abb. 42: Beispiel für ein Ausgabeschema des Listengenerators 9PAC. Die Card Number 1 ist als Control Break markiert. Da es sich dabei um die Project Number handelt, wird die Zeile T10 jedes Mal gedruckt, wenn die Zeilen eines spezifischen Projektes abgeschlossen werden sollen (im Sinne einer Summierung). Darin enthalten sind Estimated Hours (Card Number 2), Project Name (Card Number 3), Current Hours (Card Number 6) die als Accumulate, also als Summe markiert ist und Total Hours (Card Number 7), welche ebenfalls in der Spalte Accumulate als Summe markiert ist. In der Zeile D06 erscheinen die Sub-Project-Number, Current Hours, und Total Hours (des Teil-Projektes). Die Zeile D06 wird so oft wiederholt, bis ein neues Projekt beginnt und damit T10 aufgerufen wird (IBM 1961, 11).

Den Übergang von der Liste zum Netzwerk schildert Bachman am Beispiel der folgenden Transaktion: »Wenn man an eine Liste oder einer Kette von Dingen denkt, hat man eine Warenorder [als Elternknoten – F.H.], welche mit den Datensätzen der jeweiligen Einzelgegenstände [als Kindknoten] verknüpft ist. Ich möchte vom Warenorder zum ersten Einzelgegenstand gehen, dann zum zweiten Gegenstand, [...] bis zum letzten Gegenstand, welcher wieder zurück zur Warenorder verweist. Ich kann mit jedem Einzelgegenstand beginnen und wieder zur [übergeordneten – F.H.] Warenorder zurückkehren.« (ebd. 53).<sup>cxviii</sup> Information entgegen der Richtung, in der sie ursprünglich abgelegt wurde, zu lesen oder die Ebenen in den Informationshierarchien ausgehend vom aktuellen Standort zu wechseln, machte

laut Bachman das wichtigste Novum des Integrated Data Store heraus. Wichtige neue Operationen standen damit zur Verfügung: Concurrency Control (der gleichzeitige Zugriff verschiedener Nutzerinnen), Zugriffskontrolle, und die Möglichkeit, die Datenbank in einen vorherigen Zustand zu versetzen, der jenen Datenstand abbildet, der vor dem Scheitern einer Transaktion herrschte. Bachmann illustriert die Aufgabe: »Zwei Bestellungen aus zwei verschiedenen Verkaufsbüros, zwei Bestellungen aus dem gleichen Verkaufsbüro, das machte keinen Unterschied. Es handelt sich um unterschiedliche Transaktionen und man möchte, dass jede erfolgt, als wäre sie in der Datenbank komplett allein zu Gange. Im Ergebnis möchte man jedem Programm Zugang zur Datenbank geben, als gäbe es keine Überschneidung mit irgend einem anderen Programm. Das Ziel war, dass der Anwendungsprogrammierer und die Programme, welche sie schrieben, annehmen konnten, dass sie ganz allein in der Datenbank existierten, ohne Störung durch ein anderes Programm« (ebd. 88).<sup>cxix</sup> Diese Ziele wurden unter teilweiser Beteiligung von Bachman vom Team unter Leitung von Russ McGee bei Weyerhaeuser Lumbers ab 1966 in dem Nachfolge-Projekt Weyerhaeuser Comprehensive Operating System 2 verfolgt, welches ab 1968 einsatzbereit war. Ein Algorithmus sperrte spezifische Bereiche der Datenbank, auf die ein Programm zugriff, und gab sie wieder frei. War ein Bereich gesperrt, so mussten andere Programme mit der Verarbeitung warten.

Transaktion, Programm und Wartemechanismus setzte Bachman in ein Verhältnis zueinander: »In der Tat waren es die Transaktionen, die warteten, nicht die Programme. Programme sind nur Verarbeitungs-Regeln, es sind in Wirklichkeit die Transaktionen, die aktiv sind oder warten« (ebd. 90).<sup>cxx</sup> Damit betont Bachmann nicht die Hardware oder Software, sondern die Operationsketten, welche sie abbilden, die Transaktionen. Folgt man Bachmans Zentrierung der Transaktion, so lässt sich in Bezug auf die elektronische Datenbank zumindest für die 1950–1970er Jahre eine Zentralstellung der Transaktion als wichtiges Motivans für technologische Neuerungen konstatieren.



Abb. 43: Das IBM Terminal 2260, eine Bildschirm-Tastatureinheit ohne eigene Rechenkapazitäten, erlaubte den Fernzugriff auf die Ressourcen eines Zentralcomputers. Ryerson Computer Centre, ca. 1970er Jahre (Ryerson University – Library Archives, RG 63.72).

Wie groß der Bedarf an Online-Technologien wie dem Problem Controller war, zeigt sich daran, dass auch IBM 1969 seinen Anwendern die Betriebssystemnahe Softwarefiguration namens Customer Information Control System (CICS) zu Verfügung stellte. CICS war für IBM OS/360 konzipiert und ist bis heute im Einsatz. Es ermöglichte die transaktionsorientierte Delegation über Bildschirmterminals (Abb. 43) und ist tief mit anderen transaktionalen Infrastrukturen von Banken, Versicherungen oder Logistikern integriert (vgl. IBM 1972; Yelavich 1990; Campbell-Kelly 2003, 149–152). Die Leser werden dem Transaktionsmonitor CICS noch mehrfach in dieser Studie begegnen.

### 2.3.6.3 Von Stücklisten zu Kontrollsphären

Die historisch-technosoziale Dynamik der 1960er erzeugte jenes energetische Moment, welches weitere Übergänge zur Verfügung stellte. In den beginnenden 1970ern erleben wir verschiedenste Formalisierungsbemühungen jener technologischer Figurationen, die bis dahin vor allem praktisch von Ingenieuren entwickelt wurden. Einzuordnen sind diese Bestrebungen in die gleichzeitigen Bemühungen zur Strukturierten Programmierung durch Edsger W. Dijkstra (1968) und Codd's Bemühungen um eine Formalisierung von Datenbanken, die in die von Mathematik getriebene relationale Algebra mündete (1969/70). Eine weitere Formalisierung der frühen 1970er stellen Charles Davies' Überlegungen zur Kapselung von Transaktionen und die daran anschließenden Diskussionen dar. Ausgehend von der Stückliste (engl. Bills of Material), die im 19. Jahrhundert zu verorten ist, sollen weitere Grade der Formalisierung von Transaktionen in den Datenbanksystemen der 1970er Jahre diskutiert werden.

<i>Cast-iron.</i>	<i>Cubic inches.</i>
<b>40 blocks between arches, each 7½ cubic inches</b>	<b>300</b>
<b>36 post plates, each 30 cubic inches</b>	<b>1,080</b>
<b>40 coupling plates for arches, each 20 cubic inches</b>	<b>800</b>
<b>20 angle-blocks, for counter-brace rods, each 14 cubic inches</b>	<b>280</b>
<b>40 angle-blocks, for lateral brace rods, each 46 cubic inches</b>	<b>1,840</b>
<b>40 washers for lateral bolts, 3 cubic inches</b>	<b>120</b>
<b>6 angle-blocks, for diagonal brace rods, each 14 cubic inches</b>	<b>84</b>
<b>6 washers for diagonal brace rods, each 3 cubic inches</b>	<b>18</b>
<b>4 skew-backs, each 300 cubic inches</b>	<b>1,200</b>
<b>Total cubic inches of cast-iron</b>	<b>5,722</b>
<b>Weight in pounds, 1431.</b>	

Abb. 44: Stückliste für ein Element des Cove Run Viaducts, Pennsylvania (Haupt 1873, 195).

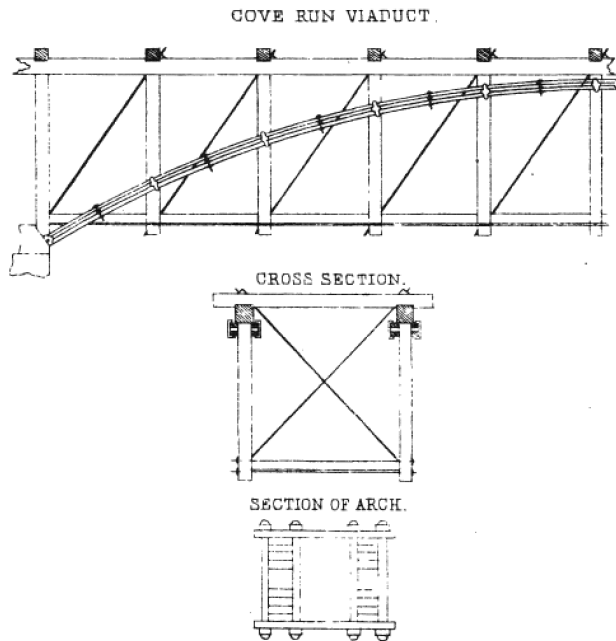


Abb. 45: Technische Zeichnung des Cove Run Viaducts mit Bezug auf die Stückliste, Pennsylvania, 1851. In diesem Beispiel sind Zeichnung und tabellarische Auflistung voneinander getrennt. In anderen Beispielen fließen diese ineinander, indem einzelne Bauteile nummeriert und somit eindeutig visuell zuordenbar werden (Haupt 1873, 271).

Stücklisten (Bills of Material) waren ein Standard-Verfahren, das aus einer Abbildung eines Gegenstandes bestand, dessen Einzelteile nummeriert waren. In einer tabellarischen Darstellung wurden diese Einzelteile entsprechend aufgeführt, sodass etabliert werden konnte, welche Einzelteile überhaupt zu produzieren waren, um beispielsweise eine Lagerwelle, ein Scharnier oder eine Brücke zu bauen (Abb. 44 und Abb. 45). Die Verwendung von Stücklisten lässt sich bis Mitte des 19. Jahrhunderts anhand von Quellen zurückverfolgen und es ist zu vermuten, dass sie den großen Festungsbauvorhaben des 17. Jahrhunderts entstammen, in deren Rahmen auch das Berufsbild des Ingenieurs entstand.<sup>189</sup> Ab Anfang des 20. Jahrhunderts wurden Bills of Material einer frühen Form der Digitalisierung unterworfen: In Lochkarten gestanzt, dienten sie der Informationsverwaltung, wie der Historiker James Wilson beschreibt. Er unterscheidet nach stabilen und instabilen, d.h. oft veränderlichen Datensätzen und zählt die Bills of Material am Beginn des 20. Jahrhunderts zu den stabilen, d.h. wenig veränderlichen Informationssammlungen. Diese ›stabilen‹ Ursprungsdatensätze konnten bei Bedarf kopiert und weiterverbreitet werden, um als Vorlagen für neue Produkte oder Teile zu dienen. Im Unterschied dazu identifiziert Wilson andere, instabilere, Informationsträger, welche für Planung, Koordination und Kontrolle der Betriebsaktivitäten zum Einsatz kamen: »Karteikarten fanden zur Koordination von Materialien aus verschiedenen Quellen an unterschiedlichen Zeiten und Orten, für eine fristgerechte Herstellung, Verwendung. Bewegungskarteikarten leiteten die Produkte von einem Prozess zum nächsten. Die Karteikarten übermittelten Rückmeldungen, indem sie zwischen Planungsbüro und Produktionshallen hin und her gesendet wurden, während die Aufgabe zuerst ausgegeben wurde, während sie die nötigen Prozesse durchlief, und ihre Fertigstellung jeweils berichtet wurde« (Wilson/James M. 2016).<sup>cxxi</sup> Die Informationsstabilität der Bills of Material sollte sich einhergehend mit der beginnenden

<sup>189</sup> Eine der frühen heute zugänglichen Quellen mit Stücklisten ist das Opus Magnum zum Holzbrückenbau durch den amerikanischen Ingenieur Hermann Haupt (1873). Für den deutschsprachigen Raum sind diese beispielsweise in Alois Riedlers Lehrbuch für Ingenieure *Das Maschinen-Zeichnen* (1896) nachweisbar.

Globalisierung und zunehmenden Diversifizierung zum Beispiel im Maschinenbau verändern. Produktvariationen und -innovationen zogen variabelere und instabilere Bill Of Materials nach sich, wie noch auszuführen sein wird.

IBMs Datenbankprodukt Information Management System (IMS, ab 1969) kannte im Unterschied zur Netzwerkstruktur des Integrated Data Store allein hierarchische Eltern-Kind-Beziehungen. Diese hierarchische Struktur verdankte Information Management System seiner Herkunft aus einem anderem Software-Produkt, dem Bills of Materials Processor (BOMP), welcher ab 1966 durch die Firmen Caterpillar und North American Aviation/Rockwell mit IBM für das Apollo-Raumfahrtprogramm entwickelt wurde und die Stücklisten des Raumfahrzeugbaus verwaltete. BOMP war auch die Basis für das DDR-Datenbanksystem BASTEI, wie noch auszuführen sein wird.

IMS ermöglichte den gleichzeitigen Zugriff auf Daten, indem Programmen erlaubt wurde, auf abgegrenzte Segmente zuzugreifen. War ein Segment belegt, mussten nachfolgende Programme warten, bis dieses freigegeben war. Was einerseits wichtig war, um überhaupt gleichzeitigen Zugriff zu ermöglichen, bereitete in der Anwendungspraxis zahlreiche Probleme, gerade bei Datensätzen mit mehreren Verknüpfungen. Erst ab 1974 war es möglich, dass Programme nicht mehr auf Ebene der Segmente, sondern einzelner Datensätze die Daten editierten, während diese für andere Programme gesperrt blieben. Realisiert wurde dieses neue Verhalten über Synchronisierungspunkte, die es erlaubten, die Softwarefiguration gegebenenfalls zurückzusetzen.

Alle Schritte bis zu einem Synchronisationspunkt wurden in einer Log-Datei aufgezeichnet, um gegebenenfalls Fehler zu finden, bzw. Transaktionen zu wiederholen. Damit konnten Transaktionen aus mehreren Einzelschritten abgebildet werden, und bei Fehlern im Rahmen einer Transaktion brach das laufende Programm ab. Ein Systembetreuer wurde informiert, um die Software auf den letzten konsistenten Zustand anhand eines Synchronisationspunktes zurück zu setzen und neu zu starten. Dafür musste die in der Log-Datei aufgezeichnete Historie von Synchronisationspunkt zu Synchronisationspunkt rückwärts abgeschrieben werden. Dies bedeutete in der Praxis, dass manuell eingegriffen werden musste und auch überprüft werden musste, an welchem Punkt die Datenverarbeitung stehen geblieben war. Die betreffenden Daten waren erneut in zu laden. (McGee 1981, 515f.).

Charles Davies, der für IBM am Information Management System arbeitete, und von dem Edgar F. Codd entscheidende Hinweise zur Maschinenunabhängigkeit von Daten erhalten hatte, kannte dessen Problematiken aus der Perspektive der Systemarchitektur in Poughkeepsie. Er wechselte 1971, ein Jahr nach der Veröffentlichung über das relationale Datenbankmodell, von der Eastcoast in das von der kalifornischen Sonne beschienene San Jose und arbeitete dort weiter an Fragen der Integrität von Datenbanken. Davies co-veröffentlichte einige Aufsätze, darunter *The Semantics of the Preservation and Recovery of Integrity for a Data System*. IBM, Mai 1972 und *Recovery semantics for a DB/DC system*, IBM 1973, in dem er das Konzept der Spheres of Control vorstellte (Abb. 46). Diese entwickelte er nicht für ein spezifisches Datenbanksystem, sondern als generelle Architektur für die Abwicklung von Transaktionen. Später erwiesen sich diese Aufsätze als einflussreich für IBMs relationalen Datenbankprototyp System R.<sup>190</sup>

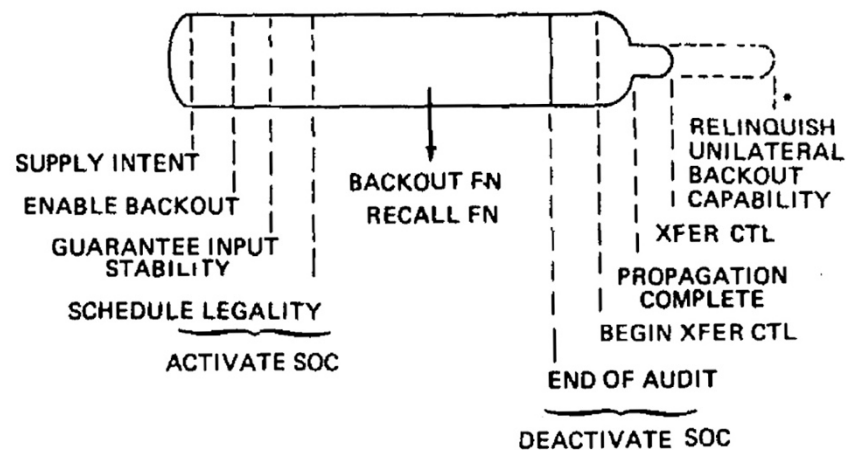
Die Recovery-Semantics, Semantiken der Wiederherstellung, sollen die Konsistenz von Datensätzen aufrecht erhalten. Die Fragestellung war, wie Datensätze, trotz einzelner fehlerhafter oder abgebrochener Eingaben, in ihrer Gesamtheit integer, unkorrupt bleiben. Davies entwickelte die Idee der Kontrollhüllen (Spheres of Control). Diese stellte eine Begrenzung dar, innerhalb derer eine Reihe von

---

<sup>190</sup> Vgl. *The Notions of Consistency and Predicate Locks in a Database System* (Eswaran u. a. 1976).



Datenbankaktionen von spezifischen Nutzern ausgeführt wurden, zum Beispiel »WÄHLE Tabelle Kundeninnen und ÄNDERE Feld Name«, und die von den anderen Nutzern abgegrenzt sind. Entweder alle Aktionen, die innerhalb einer Kontrollhülle stattfanden, schlugen sich vollständig und korrekt in der Datenbank nieder oder der Vorgang galt als unabgeschlossen und wurde in seiner Gesamtheit zurück genommen. »Eine Sphere of Control (SOC) verfügt über zugehörige Operationen: aktiviere die SOC, verlasse die SOC, rufe die SOC erneut auf, beende die SOC, übernehme eine Ressource (Nutzungsabsicht), und gib die Ressource frei (tatsächliche Nutzung)« (Davies 1973, 140).<sup>cxiii</sup>



\* First commit or release of backout guarantee.

Abb. 46: Verschiedene Abschnitte einer Sphere of Control. Sichtbar ist bereits die Konzeptualisierung mehrerer Phasen, die als Activate SOC und Deactivate SOC gekennzeichnet sind (Davies 1973, 137).

Wichtig war die Erkenntnis, dass verschiedene Nutzerprozesse nicht allein nacheinander, in Sequenz, sondern um die Rechenleistung der Maschine auszunutzen, auch gleichzeitig erfolgen sollten (Concurrency Control), beziehungsweise den Nutzern der Eindruck einer parallelen Verarbeitung erscheinen sollte. Das auf General Electrics-Computern entwickelte Information Management System und Problem Controller bzw. das daraus hervorgegangene Integrated Database Management System auf IBM OS/360-Systemen in Verbindung mit CICS, hatten hierzu bereits erste pragmatische Lösungsansätze aufgezeigt. Björk und Davies untersuchten Daten in Bezug auf die Aktionen, die an ihnen vorgenommen werden konnten und wie sich diese mit anderen Aktionen überschneiden. Können Daten geändert werden, wenn mehrere Handelnde darauf zugreifen? Müssten andere Handelnde über Änderungen in Informationsobjekten informiert werden? Kann sichergestellt werden, dass ein einmal abgelegter Wert auf Dauer unveränderlich ist, also gegen versehentliche oder ungewollte Änderungen gesichert ist? Davies konzeptueller Vorschlag bestand darin, parallel laufenden Kontrollsphären in einer übergeordneten Kontrollsphäre zu kapseln. Zu diesem Zeitpunkt wurden in der Rechentechnik bereits parallele Prozesse eingesetzt, woran es jedoch fehlte, war eine konzeptuelle Verdichtung, die Davies mit der Kontrollsphäre schuf. Dieses Konzept wurde im Zuge der Arbeiten am relationalen Prototypen System R durch Eswaran et. al. 1976 auf das Feld der Datenbanken zugespitzt und mit seinem heutigen Namen bezeichnet, als »Transaktionen«. Sie argumentierten, dass die Transaktionen sich nicht auf den physischen Teilen einer Datenbank beziehen sollten, sondern auf den logischen Teil und knüpften ausdrücklich an die Idee der Verwendung logischer Prädikate, wie wir es bei Codd gesehen haben, an.<sup>191</sup>

<sup>191</sup> Siehe auch *Principles of Transaction-Oriented Database Recovery* (Haerder/Reuter 1983, 289).

Als Konzept erhielt man mit den Spheres of Control eine eigene, spezifische Form von Historizität, eine reversible Historizität, denn die in Spheres of Control vorgenommenen Transaktionen können idealerweise komplett rückabgewickelt werden. Davies selbst war sich dieser reversiblen Historizität bewusst und formulierte in den Abschlussthesen: »Eine Anfrage, eine Transaktion, ein Auftragsschritt, ein Auftrag oder ein Tageswerk, können, sofern es nötig ist, alle in den Spheres of Control verschachtelt werden, zum Beispiel kann der Tag mit einem anderen Ergebnis erneut ablaufen, auf Kosten des Umstands, dass bis zum Tagesende keine Verpflichtungen eingegangen werden« (Davies 1973, 141).<sup>cxiii</sup> Einerseits spiegelt dies klassische bürokratische Praxen, die es erlaubten, Transaktionen wie beispielsweise Geldtransfers bei ungedecktem Konto rückabzuwickeln. Andererseits sehen wir hier eine neue Qualität des Übertragens von Daten, die gleichzeitig ein Handeln in der Realität abbildet, und die es aus der Perspektive der reversiblen Historizität weiter zu untersuchen gilt.

#### 2.3.6.4 System R – Ein relationales Forschungsprojekt

Wie treten diese informatorischen Konzepte und Modelle in die Wirklichkeit ein? Wie wurden sie implementiert und welchen Spannungen und Metamorphosen unterlagen sie im Zuge ihrer Verwirklichung? Ein gut dokumentiertes Forschungsprojekt verzeichnet die Kontroversen und Verschiebungen: System R war einer von mehreren Prototypen, die innerhalb von IBM entwickelt wurden, um das 1969 von Codd entworfene relationale Daten-Modell in einer Datenbankanwendung umzusetzen. »In großem Maße hängt die Akzeptanz und der Wert des relationalen Ansatzes an einer Demonstration, dass ein System konstruiert werden kann, welches unter realen Bedingungen eingesetzt werden kann, um reale Probleme zu lösen und das eine Leistung aufweist, die mindestens mit den heute existierenden Systemen vergleichbar ist« (Astrahan u. a. 1976, 97), schildern die Mitglieder der Forschungsgruppe ihre Aufgabe.<sup>cxiv</sup> Das Forschungsteam war ebenso wie Edgar F. Codd im IBM Forschungszentrum San Jose angesiedelt, anfangs unter seiner Leitung, später übergab Codd an einen IBM Projektmanager, da er sich weiterhin Forschungsaufgaben widmen wollte (Interview Sharon Codd 2016).

System R tritt an einem signifikanten Übergang von der Maschinen-Abhängigkeit von Daten und Transaktionen zu deren Maschinen-Unabhängigkeit auf. Nicht nur verwirklichte es die Ausformulierung (eines Teils) der Ideen Codds, es etablierte auch in Abkehr von navigatorischen Ansätzen neue Zugänge zu Informationsobjekten. Um diesen Übergang hausintern abzumildern wurde System R konsequent als IBM-internes Forschungs- und Pilotprojekt positioniert, mit der Maßgabe, die hausinternen Verfechter des bereits vorhandenen und erfolgreichen Datenbankproduktes *Information Management System* (IMS) zu besänftigen. IBM testete System R vorerst nur intern, z.B. im IBM-Werk in Owego, New York welches elektronische Flugkontrollsysteme für den LAMPS-Helikopter der US Navy produzierte, sowie bei drei ausgewählten IBM-Kunden: Pratt & Whitney, einem Hersteller von Flugzeugturbinen; Upjohn, einem Pharmazieunternehmen; und in der Flugzeugproduktion von Boeing in Seattle (Wade/Chamberlin 2012, 45).

Als wichtigen Teilaspekt bearbeitete im Anschluss an Charles Davies ein Teil des Forschungsteams jene Fragen, welche die Transaktion betrafen – gleichzeitiger Zugriff (Concurrency) und Konsistenz (bei Davies: Recovery-Semantics). Diese Vorgehensweise unterschied sich insofern von vorherigen, als dass die gesuchten Eigenschaften nicht in der Alltagspraxis einer bereits existierenden Software hinzugefügt wurden, sondern von vornherein als integraler Bestandteil von System R, ausgehend von den theoretischen Reflexionen Davies', systematisch entwickelt wurden. Nicht alle dabei eingeschlagenen Wege erwiesen sich als erfolgreich. So verfolgte man für einige Zeit die Möglichkeit, im Sinne von Codds Prädikatenlogik, die Frage des gleichzeitigen Zugriffes und der Konsistenz deklarativ zu lösen (Eswaran u.

a. 1976), musste schließlich diesen Ansatz aus Gründen der Rechenleistung und Komplexität verwerfen und entwickelte eine prozedurale Vorgehensweise (Chamberlin et. al. 1981, 639). Dem Aufsatz *The Notions of Consistency and Predicate Locks in a Database System* von Kapali P. Eswaran, James N. Gray, Raymond A. Lorie, and Irving L. Traiger entstammt jedoch das grundlegende Konzept der Atomizität, welches wie folgt hergeleitet wird: »Der Systemzustand ist nicht statisch. Er unterliegt fortwährenden Änderungen aufgrund der Aktionen, welche von Prozessen an Entitäten [Informationsobjekten – F.H.] durchgeführt werden. Lesen und Schreiben sind Beispiele für solche Aktionen« (Eswaran u. a. 1976, 624).<sup>cxv</sup> Eine Transaktion wurde als eine Folge von Aktionen beschrieben, welche Änderungen verursachen.

Auffallend ist, dass von vornherein das Datenbankmanagementsystem nicht als statisch, sondern als dynamisch gedacht wurde. Die beständigen Änderungen erzeugten eine Situation, in der nicht allein die Informationsobjekte dynamisch sind, sondern auch die Figuration selbst, welche die Informationsobjekte verwaltet. Permanent passt sich das Datenbankmanagementsystem dem Wandel im Datenkorpus an – zum Beispiel indem Indexe aktualisiert werden, indem abhängig von der Komplexität der Abfragen (Queries) unterschiedlicher Maschinencode der in den Tiefen der Maschine agiert, generiert wird, oder indem die Zugriffe verschiedenster Nutzer, die sich teils gleichen können, und teils sehr unterschiedlich sein können (Chamberlin et. al. 1981), entsprechend abgebildet werden. Um diesem Anspruch gerecht zu werden, wurde definiert, dass die Ausführung von Anfragen an den Datenbestand einer Reihenfolge unterliegen soll: »Wir gehen davon aus, dass diese Aktionen atomisch sind; das bedeutet, wenn zwei Prozesse gleichzeitig Aktionen ausführen, ist die Auswirkung als ob eine Aktion vor der anderen ausgeführt wurde«(Eswaran u. a. 1976, 624).<sup>cxvi</sup>

Bereits in Bachmans Integrated Data Store war die Frage aufgeworfen worden, auf welcher Ebene Datensegmente im Zugriff der verschiedenen Rechenprozesse voneinander abgegrenzt werden sollen. In System R entschied man sich für einen granularen Ansatz: »Dieses Schema beinhaltet eine Hierarchie von Sperrern (engl. ›Lock‹) verschiedenen Umfanges, die von einzelnen Datensätzen zu mehreren Tabellen reichen« (Chamberlin et. al. 1981, 639).<sup>cxvii</sup> Untergeordnete Sperrern eines Datenbereiches, z.B. ein Datensatz, ›reservierten‹ ihre übergeordnete Bereiche, zum Beispiel die Tabelle, in der sich der spezifische Datensatz befand. Durch diese Sperrern und Reservierungen konnten verschiedene Nutzer gleichzeitig auf unterschiedliche Datensätze in einer Tabelle zugreifen, ohne Gefahr zu laufen, gegenseitig Daten zu überschreiben. Eine Sperre auf Tabellenebene, zum Beispiel wenn der Tabellenstruktur eine neue Spalte hinzugefügt werden sollte, konnte erst erfolgen, wenn alle untergeordneten Sperrern aufgehoben waren, das heißt, sobald die Prozessierung der jeweiligen Datensätze beendet war (ebd.). Dieser in den 1970ern erarbeitete Komplexitätsgrad der granularen Verarbeitung von Informationsobjekten ist bis in die 2000er Jahre wirksam geblieben, bis neue Paradigmen wie Googles' Bigtable oder Post-SQL diesen Ansätzen weitere Zugriffsmethoden hinzufügten.

Weiterhin verfolgte das Team intensiv die Frage, wie ein konsistenter Zustand der Datenbank hergestellt werden könnte, wie also ein Zustand erreicht würde, indem alle Transaktionen abgeschlossen sind. Dabei war spätestens seit Davies klar, dass Konsistenz immer nur zu bestimmten Zeitpunkten erreicht werden würde, an einem ›Checkpoint‹. Der Dauerzustand relationaler Datenbanken ist Inkonsistenz, die sich aus den ständig ablaufenden, sich überschneidenden Transaktionen und deren atomischen Aktionen ergibt. System R war dazu ausgelegt, immer dann, wenn gerade alle laufenden Transaktionen der verschiedensten Nutzer und Prozesse abgeschlossen waren, den für jenen kurzen Moment konsistenten Zustand festzuhalten und einen Checkpoint zu setzen. Dieser konnte zur Wiederherstellung bei Abstürzen, Lese-/Schreibfehlern von Datenträgern oder Abbrüchen innerhalb von Transaktionen aufgesucht werden.

Die Mikro-Historizität der Transaktion ist einerseits im Log eingeschrieben, welches die jeweils zuletzt ausgeführten Transaktions-Atome (z.B. die Befehle INSERT, DELETE, UPDATE) enthielt. Hinzu kam andererseits die Shadow Page-Technik, welche als Schatten, den die Datenobjekte warfen, einen alten und einen aktuellen Zustand der Informationsobjekte abzubilden erlaubte. Während Transaktionen prozessiert wurden, definierten die ›alten‹ Shadow Pages den letzten verfügbaren konsistenten Zustand. Stürzte die Software ab, konnte sie nach einem Neustart anhand der ›alten‹ Shadow-Pages zurückgesetzt werden. Anhand der Logaufzeichnungen konnten die abgebrochenen Transaktionen wiederholt werden (Astrahan u. a. 1976, 123f.; Chamberlin et. al. 1981, 639; McGee 1981, 516). Neu war auch, dass die Nutzer bzw. Programme Datenbank-Transaktionen in der neu entwickelten Sprache SEQUEL unmittelbar starten und beenden konnten, durch die Befehle BEGIN TRANS und END-TRANS. Inmitten einer Transaktion wurden mit SAVE spezifische Rückkehrpunkte festgelegt, die mit Hilfe von RESTORE wiederhergestellt werden konnten (Astrahan u. a. 1976, 108). Diese Rückgängig-Funktion erlaubte das Zurückspulen von Zeit und ein erneutes Anstoßen von Transaktionen. Um wiederholende Transaktionen, wie beispielsweise Banküberweisungen, zu optimieren, kamen wiederverwendbare ›Views‹ zum Einsatz. Üblicherweise wurde für jede Abfrage der ausführbare Maschinencode neu optimiert und kompiliert und die optimalen Zugriffspfade auf die Daten ermittelt. Patricia Selinger entwickelte diese Komponente – den Optimizer – maßgeblich (Selinger u. a. 1979). Für sich wiederholende Transaktionen, wie Banküberweisungen, gab es jedoch einen effizienteren Weg: Mit Hilfe von Views konnten Abfragen einmal definiert und durch den Optimizer vorbereitet und zwischengespeichert werden. In der Folge konnten sich wiederholende Transaktionen durch die gespeicherten Views aufgerufen werden, wobei der Umweg über den Optimizer eingespart wurde. Dies ermöglichte es, sowohl Ad-Hoc-Abfragen, als auch regelmäßig wiederkehrende Transaktionen gleichermaßen zufriedenstellend zu verarbeiten – ein wichtiger Erfolgsfaktor für die Durchsetzung relationaler Datenbanksysteme (Wade/Chamberlin 2012, 44).

Die Arbeit an dem Prototypen führte auch zu der Erkenntnis, dass eine Maßgabe des Optimizers nicht die Anzahl der verarbeiteten Tupel wäre, sondern die Anzahl der Eingabe/Ausgabe-Operationen – eine Einsicht, die zukünftig noch für die Bewertung der Transaktionsperformance produktiv werden sollte (Chamberlin et. al. 1981, 634).

Diese ausführliche Darstellung wurde unternommen, um aufzuzeigen, mit welcher Intensität im Zuge von System R Methoden der Transaktionsverarbeitung erforscht und implementiert wurden, die, trotzdem sie in die maschinellen Tiefen des Computers reichen, für die Nutzer transparent blieben. In der Kombination all dieser Methoden vollzieht sich der Übergang zu einem neuen Begriff der Transaktionen, welcher bereits in den Pragmatiken des am Netzwerk-Modell orientierten Integrated Data Store und IBM's hierarchischen Information Management System schlummerte: Waren mit ›Transaktion‹ bisher die von Menschen in Verbindung mit mechanischen Systemen durchgeführten Aktionen gemeint, welche in der Rechenmaschine modelliert und nachvollzogen wurden, übersiedelte der Begriff ›Transaktion‹ nun in das Feld der Informationsverarbeitung: »[...] die Aktionen eines Prozesses sind in Sequenzen gruppiert, welche als Transaktionen bezeichnet werden, und welche Einheiten von Konsistenz sind« (Eswaran u. a. 1976, 624).<sup>cxxviii</sup>

In-Formatisierte Transaktionen zeichnen sich dadurch aus, dass sie selbst einen Übergang darstellen – den Übergang von einem konsistenten Zustand zum nächsten. Die einzelne Transaktion wurde dabei in zwei wesentliche Phasen unterteilt: »[...] Konsistenz erfordert, dass eine Transaktion in eine wachsende und in eine schrumpfende Phase unterteilt ist. Während der wachsenden Phase ist es der Transaktion erlaubt, Sperren zu setzen. Die schrumpfende Phase beginnt erkennbar mit der ersten

Aufhebung einer Sperre« (Eswaran u. a. 1976, 627).<sup>cxxxix</sup> Je mehr in der zweiten Phase Sperren aufgehoben werden, desto mehr nähert sich die Transaktion ihrem Ende und die Gesamtfiguration einem Zustand der Konsistenz an. In diesen Punkten gingen die Ergebnisse der Arbeit am Prototypen System R über die von Codd entwickelten Vorschläge hinaus, verwiesen in eine andere, pragmatischere Richtung und bildeten die Grundlage für viele andere relationalen Datenbanksysteme.

#### 2.3.6.5 Technologietransfere und ›Benchmark Wars‹

Das IBM Team publizierte viele der Erkenntnisse in Fachzeitschriften und machte sie damit anderen Entwicklern zugänglich, wie sich in einer wachsenden Vielfalt relationaler Produkte in den 1980er Jahren zeigte.<sup>192 cxxx</sup> IBM selbst verfolgte nach dem Ende der Projektgruppe System R im Dezember 1979 zwei relationale Datenbanksysteme, zum einen Structured Query Language/Data Store, kurz SQL/DS (1981), für die Betriebssysteme DOS/VSE und VM/CMS und zum anderen mit deutlich größerer Verbreitung DB2 für das MVS Betriebssystem (ab 1983 verfügbar).<sup>193</sup>

Bereits während der Forschung an System R waren andere Entwickler in Erscheinung getreten, um die relationale Algebra in einem Datenbankprodukt umzusetzen. Darunter waren am wichtigsten die Datenbankmanagementsysteme INGRES an der Berkeley University und Oracle.<sup>194 cxxxi</sup> Beide stellten Eigenentwicklungen dar, die auf jeweils spezifische Weise im Austausch mit System R standen.

INGRES war ein universitäres Forschungsprojekt. Im Frühjahr 1972 untersuchte eine Seminargruppe verschiedene Datenbankmodelle und entschied sich, den relationalen Ansatz anhand des von Codd publizierten Fachaufsatzes zu verfolgen.<sup>195</sup> Als sich abzeichnete, dass an der Universität die Forschungsarbeit in andere Bereiche führen sollte (u.a. CAD-Anwendungen), gründeten die beteiligten Lehrenden Michael Stonebraker und Eugene Wong eine Firma aus, welche die weitere Entwicklung vorantrieb.

---

<sup>192</sup> Siehe die umfangreichen Quellenangaben in *A History and Evaluation of System R* (Chamberlin et. al. 1981, 646), welche 48 von insgesamt 50 Positionen umfassen, die von IBM-Team Mitgliedern in den Jahren 1970 bis 1980 in einschlägigen Fachzeitschriften publiziert wurden. Zieht man jene 11 Publikationen ab, die als IBM Research Report eine vergleichsweise eingeschränkte Verbreitung fanden, so erzeugte IBMs Forschungsprojekt innerhalb von 10 Jahren 37 Veröffentlichungen, die alle Aspekte eines relationalen Datenbankmanagementsystems betrafen: »Im Unterschied zu vielen Firmen, unterhält IBM einen Forschungsabteilung mit Laboratorien, welche von ihren Produktentwicklungsabteilungen getrennt sind. In der Zeit von System R bestand die IBM-Forschung aus drei Laboratorien, gelegen in Yorktown Heights, in San Jose und in der Nähe von Zürich. [...] Alle IBM Forschungslaboratorien haben eine doppelte Rolle: einen Beitrag zum Voranschreiten der Wissenschaft zu leisten und zukünftige Technologien für IBM zu entwickeln. Ähnlich Universitätsprofessoren sind die IBM Forschungs-Mitarbeiter angehalten, an technischen Konferenzen teilzunehmen und sie werden teilweise basierend auf ihren Veröffentlichungen in peer-reviewed Veröffentlichungen evaluiert« (Wade/Chamberlin 2012, 43).

<sup>193</sup> Vgl. dazu *The History and Growth of IBM's DB2* (Haderle/Saracco 2013).

<sup>194</sup> Die personellen Verbindungen zwischen System R und INGRES sind vielfältig. Zum einen hatte Edgar F. Codd zu einem Zeitpunkt, als unklar war, ob der relationale Ansatz innerhalb IBMs durchsetzungsfähig war, durch Vorträge und Besuche an Universitäten auch im Austausch mit INGRES gestanden, vor allem im Jahr 1974 (Jerry Held In: Batti et. al. 2007, 20). Zum anderen waren mehrere System R Mitarbeiter (u.a. Jim Gray, Bruce Lindsay, and Randy Katz) in dem Studiengang ausgebildet worden, in dem später auch INGRES entwickelt werden sollte (Wade/Chamberlin 2012, 44; McJones 1997, 11f.). Die Entwicklung von Oracle basierte grundlegend auf den Publikationen über System R und auch auf der Übernahme von IBM-Mitarbeitern: »Die Leute von SDL (später Oracle Inc.) waren weder Akademiker, noch mathematische Theoretiker. Sie hätten SQL nicht selbst entwickeln können. Doch sobald IBM das erledigte, wussten sie, was zu tun war. Jetzt war die Aufgaben, ein DBMS [Datenbankmanagement-system] zu bauen, als ob man ein Modellflugzeug aus einem Fertigteilset konstruierte« (Wilson in: McJones 1997, 64).

<sup>195</sup> An anderer Stelle sind die historischen Ereignisse ausführlicher dargestellt wurden. Die Entwicklung von INGRES ist beschrieben in (Stonebraker/Michael R./Wong/Held 1975; Stonebraker/Michael u. a. 1976; Stonebraker/Michael 1980; Rowe 2012) und eine Darstellung von Oracle bieten (Wilson/Mike 1997; 2007; Preger 2012; Mendelsohn 2013). Der vorliegende Text verfolgt jene Aspekte, die Transaktionen betreffen.

»Um 1978 war klar, dass ein relationales System signifikante Vorteile gegenüber der konkurrierenden DBMS-Technologie hatte, also war absehbar, dass die Technologie in Produkte übertragen würde« (Rowe 2012, 70).<sup>cxviii</sup> Während IBM durch seine Präsenz in den USA und international die Möglichkeit hatte, die eigenen Datenbanksysteme auf Mainframe-Computern wie IBM System 360/370 zu verbreiten, kam dem Projekt INGRES das Aufkommen kleinerer Computer, wie DEC PDP, VAX und später SUN Microsystems, ab Anfang der 1980er, zu Gute. Diese wurden aufgrund geringerer Anschaffungskosten für die Lehre eingesetzt. Als Betriebssystem verwendeten sie UNIX.

Die Firma AT&T, ein Monopolist im Telekommunikationssektor, stellte das ab 1969 entwickelte UNIX als Open-Source zur Verfügung (Ceruzzi 2003, 282–285). Es verbreitete sich massiv an amerikanischen Universitäten, und das universitäre INGRES erwies sich als das dazu passende Open-Source Datenbanksystem. Ebenso wie System R suchte INGRES zwei Stärken miteinander zu vereinen: Zum einen sollten Ad-Hoc-Abfragen schnell ausgeführt werden und auch von Nicht-Programmierern zu erstellen sein, zum anderen sollte Transaktionssicherheit und -geschwindigkeit geboten werden. So rahmte, in einem der frühen veröffentlichten Forschungspapiere über INGRES, Michael Stonebraker die Ansprüche an Transaktionen als eine Frage der Effizienz und hält fest, dass Ingres in seiner Ursprungsfassung Mitte der 1970er Jahre in der Lage war, 2,5 Transaktionen pro Sekunde zu verarbeiten (Stonebraker/Michael 1980, 237). Dies konnte mit damals aktuellen Mainframe-Systemen nicht mithalten, welche zwischen 200 und 400 Transaktionen pro Sekunde abwickeln konnten.

Als Flaschenhals erwies sich der Optimizer und das ab 1980 kommerzielle INGRES stellte schließlich den Berkeley Absolventen Bob Kooi ein. Kooi hatte im Rahmen seiner Dissertation den von Patricia Selinger für System R entwickelten Optimizer untersucht und weiter entwickelt. Unter seiner Leitung gelang es, mehrere Technologien erfolgreich nach INGRES zu transferieren: »Wir hatten die Idee, die ersten zu sein, die 100 TPS [Transactions Per Second – F.H.] auf einem Relationalen Datenbankmanagementsystem demonstrieren konnten. Dieses Ziel wurde 1988 erreicht [...]« (Rowe 2012, 67).

Die Erzählung immer steigender Transaktionszahlen pro Sekunde ist exemplarisch kennzeichnend für jene Aushandlungsprozesse, die am Übergang aus dem Forschungsstadium in technologische Produkte beobachtbar sind. Den Industrieexperten war die Prekarität der Erzählung ›Transaktionen pro Sekunde‹ frühzeitig bewusst. Ihre Kunden, z.B. Rechenzentren in Versicherungen oder Banken, benötigten jedoch Kriterien anhand derer sie sich für eine spezifische Hardware-Software-Figuration zur Transaktionsdatenverarbeitung entscheiden konnten. Und das Marketing der Hersteller benötigten Argumente, um sich von der zunehmenden Konkurrenz relationaler Datenbankmanagementsysteme deutlich zu unterscheiden: Transaktionen pro Sekunde (TPS) wurde eines der Kriterien.

1980 hatten Paula B. Hawthorn, damals Vize-Präsidentin der Computerfirma Britton-Lee, welche dezierte Computer für Datenbanken produzierte und der Assistant-Professor David J. DeWitt an der University of Wisconsin verschiedene »Datenbankmaschinen« untereinander verglichen, um eine Hardware-orientierte Einschätzung über die Geschwindigkeit der Transaktionsverarbeitung abgeben zu können (Hawthorn 2002). In den folgenden Jahren wurde eine Reihe von Studien publiziert, welche eines gemeinsam hatten: Die aufstrebende und teils aggressive agierende Datenbank-Firma Oracle stach durch relativ lange Abfragezeiten heraus.

System	Number of Tuples Selected from 10,000 Tuple Relation	
	100	1000
U-INGRES	53.2	64.4
C-INGRES	38.4	53.9
ORACLE	194.2	230.6
IDMnodac	31.7	33.4
IDMdac	21.6	23.6
DIRECT	43.0	46.0
SQL/DS	15.1	37.1

Abb. 47: Select-Abfragen ohne Verwendung eines Index; Zeit in Sekunden für 100 und für 1.000 Tupel aus 10.000 in der Datenbank abgelegten Tupeln. Eine von mehreren Übersichtstabellen in den Oracle schlecht abschnitt (Bitton/DeWitt/Turbyfill 1983, 15).

Nachdem rechtliche Scharmützel nicht die gewünschten Ergebnisse brachten, stellte Oracle schließlich den Entwickler Bob Kooi vom Konkurrenten INGRES und vormals System R eins, um den Optimizer neu zu entwickeln (Rowe 2012, 63). Genealogisch kam damit die von Patricia Selinger für System R entwickelte Optimizer-Technologie in einer Reihe von relationalen Datenbankprodukten zur Anwendung: In IBMs SQL/DS, in INGRES und in Oracle.<sup>196</sup> Laut Einschätzung von DeWitt triggerte die Veröffentlichung eine Reihe von ›Benchmark Wars‹, in deren Zuge die Leistung in Transaktionen pro Sekunde anzugeben versucht wurde. Das Benchmarking sorgte in der Konsequenz für eine Angleichung der Hersteller, da die schlechter abschneidenden Anbieter verstärkt in ihre Software investierten (DeWitt In: Gray (Hrsg.) 1991, 2).

Der große Einfluss des System R Optimizers soll an einem weiteren Beispiel gezeigt werden: Jim Gray und Gianfranco Putzolu waren 1980 von IBM Datenbankforschung zur Mitte der 1970er Jahre gegründeten Firma Tandem gewechselt, die sich auf Server-Systeme für die Abwicklung von Transaktionen spezialisiert hatte.

Mit ihnen ging auch das Wissen aus der Entwicklung von IBM System R und IBM/DB2 zu Tandem über. Ihre Ambitionen manifestierte sich in der Systemprogrammiersprache Transaction Application Language und in einem Betriebssystem namens Guardian, welches versprach, vor Datenverlusten zu schützen. In der Produktdokumentation zeigte sich die enge Anlehnung an den Begriff der Transaktion, wie er von Gray et. al. 1981 geprägt wurde: »Eine Transaktion ist definiert als eine Folge von Einfügungen, Modifikationen oder Löschungen, die an Datensätzen an einzelnen Knoten oder multiplen Knoten in einem Netzwerk zwischen den Systemaufrufen ›begin-transaction‹ und ›end-transaction‹ erfolgt sind, und die eine Datenbank von einem konsistenten Zustand in den nächsten überführen« (Schuster/Tandem Computers 1981, 11).<sup>cxxxiii</sup>

Das Neue an dem von Gray bei Tandem mit-entwickelten Folgeprodukt NonStop-SQL (Abb. 48) war die Fähigkeit, mehrere Prozessoren parallel zu verwenden, bei gleichzeitiger maximaler Ausfallsicherheit und Skalierbarkeit. Transaktions-Ressourcen konnten modularisiert werden, und die Verwendung von Netzwerken (1981 war der Standard ›Ethernet‹ von einem Industriekonsortium festgelegt wurden) war bereits konzeptuell im Produkt angelegt.

<sup>196</sup> Selingers Optimizer wurde auch in der DDR ausgiebig untersucht und die Prinzipien in eigene Software übernommen, siehe Kapitel 3.3 *Kalifornien, Technische Universität Dresden und Robotron Dresden: Das Problemorientierte Seminar »Datenbanken« und DABA 1600.*

*“The Union Bank of Finland handles 100% of retail banking services in 513 locations on a Tandem NonStop™ Computer Network.”*

Risto Wartiovaara  
First Vice President  
Union Bank of Finland Ltd.

In addition to supporting Union Bank of Finland's nationwide network of 2300 teller terminals, our Tandem NonStop computer serves 105 on-line ATMs with less than one second internal access time. The bank has already achieved throughput of more than 45 retail transactions per second, with even more anticipated for the future.

Because we go back a long way with data processing systems, we can really appreciate these kinds of state-of-the-art features in a computer system. As well as the system's many other performance benefits that are absolutely essential for today's electronic banking services.

Like data integrity. And an easy growth path for future expansion. No other mainframe manufacturer was able to meet our requirements for reliability and growth in such a simple, straightforward way. Not only that, Tandem's was by far the most cost-effective solution.

Thanks to Tandem's comprehensive set of easy-to-use programmer tools, application development was remarkably smooth for an operation of such national scope; the system went on-line exactly on schedule and without the slightest problem.

The future? The system already in place is a totally integrated transaction processing network. All we have to do to expand is add on new applications in low-cost increments.

**The NonStop System.** The only system on the market today that can provide a dispersed network of up to 255 systems, each ranging from 2 to 16 processors, supporting thousands of terminals in an on-line, NonStop transaction processing environment.

Tandem. Fully supported by a worldwide sales, training, service and manufacturing organization.

For more information on the Tandem application at Union Bank of Finland, and a copy of our brochure, "Solutions in Banking", contact your local sales office or Tandem Computers Incorporated, 19333 Valico Parkway, Cupertino, California 95014, U.S.A. Toll Free 800-538-3114 or (408) 725-7500 in California.

**TANDEM**  
NonStop Transaction Processing

Abb. 48: Beispiel für die Werbung mit Fokus auf Transaktionen. Werbung für Tandem-Computers NonStop System: »Bis zu 45 Retail-Transaktionen pro Sekunde« (Tandem Computers 1983, 34–35).

Die Passage von raumfüllenden, stapel-verarbeitenden Mainframe-Computern zu kleineren, vernetzten Server-Client-Einheiten kennzeichnet die technologische Dimension; das Bemühen um die Vergleichbarkeit verschiedenster Transaktionsfigurationen verweist auf die delegierende und koordinierende Dimension. Benchmarks korrespondierten mit dem Ziel, sich auf Standards zu einigen und waren Teil jener infrastrukturellen Arbeit, die letztlich für eine Verstetigung neu entwickelter Technologien sorgt. Dies ruft die Argumentation von Ruhleder/Star auf, welche die Emergenz von Infrastrukturen als zeitlichen Prozess hervorheben. Nicht die Frage, »Was ist eine Infrastruktur«, sondern »Wann ist eine Infrastruktur«, sei zu verfolgen (Star/Ruhleder 1996, 4f.). Für die Transaktion kann dieses ›Wann?‹ einerseits an den ›Benchmarking Wars‹ festgemacht werden und andererseits an der Durchsetzung des ACID-Prinzips, welches im Folgenden, letzten Abschnitt dieses Kapitels dargestellt wird.

#### 2.3.6.6 Standardisierungen durch ACID

Das anschließend an Gray et al. durch Theo Härder und Andreas Reuter 1983 eingeführten Akronym ACID kennzeichnete Transaktionen im informatorischen Sinne durch vier Eigenschaften: Atomizität, Konsistenz, Isolation, und Dauerhaftigkeit. Dass es sich um einen Ansatz zur Konsolidierung handelt, wird gleich im ersten Satz ihres Aufsatzes *Principles of Transaction-Oriented Database Recovery* deutlich. Die Transaktion bezeichnen die Autoren als »das wesentliche Paradigma für Synchronisierung und Wiederherstellung in fortgeschrittenen Datenbanksystemen« (Haerder/Reuter 1983, 288).<sup>cxxxiv</sup>



Eine Transaktion mache aus, dass sie unteilbar sei und Unteilbarkeit werde erreicht durch:

- Atomizität – es handelt sich um eine Alles- oder Nichts-Aktion,
- Konsistenz – die Beendigung von Transaktionen garantiert, dass die Daten widerspruchsfrei sind,
- Isolation – die Aktionen verschiedener Transaktionen sind voneinander abgegrenzt, sodass sie sich nicht gegenseitig überschneiden, und
- Dauerhaftigkeit – Sobald eine Transaktion abgeschlossen ist, muss garantiert sein, dass bei Fehlern oder Abstürzen die Informationsobjekte dauerhaft erhalten bleiben.<sup>197</sup>

Aus diesen vier Eigenschaften ließe sich laut Haerder/Reuter die Qualität eines Datenbankmanagementsystems herleiten. Ihr Anspruch war es, eine Taxonomie zu liefern, die es erlaubte, aktuelle Datenbankmanagementsysteme in Bezug auf ihre Fähigkeiten zur Datenkonsistenz- und Wiederherstellung zu klassifizieren, mit dem Ziel, eine Methodologie als Entscheidungshilfe für Nutzer und Käufer zur Verfügung zu stellen.

ACID wurde für über 20 Jahre als Industriestandard für transaktionsorientierte (relationale) Datenbanken maßgebend. Erst in den seit Mitte der 2000er expandierenden Big-Data Technologien wurde die Konsistenz von Datenbeständen neu gewichtet, zugunsten der Möglichkeit, riesige Mengen an Daten in verteilten Rechenzentren zu sammeln. Zu den neuen Semantiken zählte seit 1999 das CAP-Theorem (Consistency, Availability, Partition Tolerance), welches zugunsten einer besseren Verteilbarkeit der Datenbanken über Netzwerke plädierte (Fox/Brewer 1999). Das Konzept BASE (Basically Available, Soft state, Eventual consistency) forderte nicht mehr unmittelbare Konsistenz ein, sondern versprach, zeitverzögert in Richtung Konsistenz zu streben (Terry u. a. 1995; Pritchett 2008), um der stärkeren globalen Verteilung von Daten und der weiter wachsenden Datenmengen gerechter zu werden. Diese neuen Semantiken der Konsistenz gingen einher mit der Verbreitung post-relationaler Datenmodelle wie Google BigTable und Mapreduce seit 2004, Amazon SimpleDB seit 2007, und der Open-Source-Variante Apache Hadoop/Mapreduce seit 2011. Es wäre allerdings falsch anzunehmen, dass die neuen Konzepte ACID abgelöst hätten, sie stellen vielmehr neue Konzepte für delegierende, verteilte Figuren dar, die nicht notwendigerweise jederzeit konsistent sein müssen, beispielsweise ›Soziale‹ Medien Anwendungen. Davon unabhängig folgen Schätzungen zufolge 75% der heute im Produktiveinsatz befindlichen Datenbankmanagementsysteme weiterhin dem ACID-Paradigma relationaler Datenbanken (Stand 2020, DB-Engines/Solid IT 2020).

In diesem Abschnitt haben wir gesehen, wie die Online-Transaktionsverarbeitung (OLTP) aus Techniken des nichtlinearen Speicherzugriffs und der Parallelisierung von Datenzugriff und -verarbeitung entstanden ist. Wir haben gesehen, wie die Wissenstechnik ›Bills of Material‹ die Notwendigkeit der Konzeptualisierung von Transaktionen als Kontrollsphären nach sich zog und wie Transaktionen die Entwicklung von IBMs einflussreichem System R vorantrieben. Transaktionen standen außerdem im Mittelpunkt der ›Benchmark-Kriege‹ der 1980er Jahre, bis sie unter dem Akronym ACID standardisiert wurden. Damit ist die Grundlage geschaffen, um im nächsten Abschnitt zu untersuchen, wie die Verarbeitung von Transaktionen zu einem Ausgangspunkt für neue Ansätze der Datenanalyse wurde.

---

<sup>197</sup> Castelle verweist darauf, dass Atomizität, Isolation und Dauerhaftigkeit in verschiedensten Systemen vergleichsweise formal implementiert werden können, während die Erzeugung von Konsistenz jeweils den unterschiedlichen Anwendungen entsprechende Semantiken benötigt, und damit stärker von der Arbeit der Nutzer der jeweiligen Datenbank abhängig ist, als die anderen drei Kriterien (Castelle 2013).

### 2.3.7 On-Line Analytical Processing (OLAP)

Theoretisierung, Produktentwicklung, Infrastrukturalisierung und Standardisierung bereiten den Weg für die relationale Datenbank als Vermittlerin zwischen Betriebssystem und den Datenbedürfnissen ihrer Nutzer an der Oberfläche des User-Interfaces. Sie wurden Wegbereiter neuer Verfahren zur Analyse von Informationsobjekten. Diese neuen Verfahren der späten 1980er können auf lange Genealogien zurückblicken, ob man nun als einen der Ausgangspunkte die doppelte Buchhaltung oder die tabellarischen Verfahren der Wissensorganisation und -generierung (und weitere, hier nicht dargestellte) nimmt.

Die maschinelle Verarbeitung von Informationen und Transaktionsdaten ermöglichte eine Ausweitung und Beschleunigung der Informationsanalyse in den letzten Jahrzehnten. In den frühen 1990er Jahren las sich die Motivation für automatisierte Analysewerkzeuge so: »Eine wachsende Anzahl von Managern denken in folgender Weise: ›Wir haben die operationalen Daten unter Kontrolle, den Kern unseres Geschäfts, in dem wir unsere Bestellungen nachverfolgen. Jetzt wollen wir ein System aufsetzen, um Erkenntnisse aus dem strategischen Wert der Daten zu gewinnen. Wer sind unsere Kunden und wie sollten wir unsere Märkte segmentieren? Welche unserer Kunden generieren den größten Gewinn? Welche Trends zeichnen sich in Produktverkäufen ab? Können wir unser Wissen unserer Kundenbedürfnisse einsetzen, um sie zu beraten, mehr unserer Produkte zu liefern und gleichzeitig ihre Wahrnehmung unseres Werts den wir für sie haben verbessern, bevor unsere Konkurrenz uns darin schlägt?« (O’Neil In: Gray (Hrsg.) 1991, 1).<sup>cxv</sup> So beschrieb Patrick E. O’Neil (University of Massachusetts, Boston) Anfang der 1990er Jahre die Anforderungen an eine Reihe von Anwendungsprogrammen, welche er als »Marketing Informationssysteme, Entscheidungsfindungssystem, Management-Berichtserstattung oder Direktmarketing« zusammenfasst (ebd.).<sup>cxvi</sup> Gemein ist diesen Anwendungen, dass sie nicht Transaktionen abwickeln, sondern einen analytischen Umgang mit Transaktionsdaten ermöglichen. Dafür wurde in der ersten Hälfte der 1990er Jahre das Akronym OLAP entwickelt, welches jene Werkzeuge beschreibt, die ein On-Line Analytical Processing ermöglichen. Wichtig für OLAP ist dessen hier aufgezeigte Genealogie aus den bürokratisch-wissensorientierten Feldern der Informationsverarbeitung, also nicht allein aus vordergründig kybernetisch motivierten Kontrollphantasien – wenn auch Ideen der Operations Research einfließen.

Das Ziel von OLAP war die Zusammenführung organisationsweiter Daten aus uneinheitlichen Quellen in einer vereinheitlichten Datenstruktur, ohne die zugrundeliegenden Operationen des OLTP einzuschränken. Die operationalen Daten in der Datenbank sollten aus Performancegründen unberührt bleiben. Daher wurden in OLAP-Figurationen diese Daten von der eigentlichen Quelle abgetrennt und gedoppelt. Diese Entkopplung diente dazu, Operationen zu umgehen, die in den Ursprungsdatenbanken die Performance beeinflussen konnten, beispielsweise das Sperren im Zuge der Concurrency Control. Die Doppelung der Daten diente aber auch dazu, Zustände festzuschreiben, die später als zeitlicher Abstand in Wert gesetzt werden konnten. Als Begriff für diese Vereinheitlichung und das Vorhalten von großen Datenmengen etablierte sich ebenfalls Anfang der 1990er Jahre *Data Warehouse* (Abb. 49) (Inmon 1992).

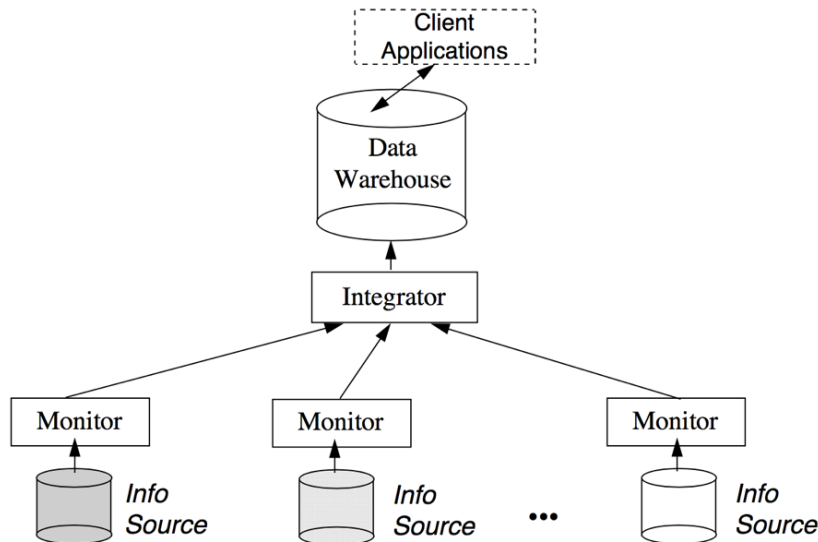


Abb. 49: Schema für ein Data Warehouse. Im unteren Bereich befinden sich verschiedenste dynamische Datenquellen, aus denen Monitorprogramme sich verändernde Daten kopieren. Der Integrator dient der Vereinheitlichung unterschiedlicher Formate, bevor diese ins Data Warehouse übernommen werden. Auf das Data Warehouse greifen verschiedenste Anwendungen zur Visualisierung und Prozessierung zu (Hammer u. a. 1995).

Die Vereinheitlichung von Datenformaten und -modellen als Zwischenschritt erwies sich als notwendig, da beispielsweise Datumsangaben an Point-Of-Sales in unterschiedlichen Formaten vorliegen konnten und sich vom Datumsformat in der Kundendatenbank eines Unternehmens unterschieden. Die Zusammenführung sollte sicherstellen, dass die Referenz der vereinigten Datenobjekte gleich blieb. Dabei ging es nicht, wie in OLTP, um eine schnelle Abwicklung vieler kleiner Transaktionen und Ausfallsicherheit, sondern um einen analytischen Blick auf große Mengen von Transaktionen aus verschiedenen Perspektiven. OLAP erweist sich dabei als ein taktisches Werkzeug, welches Manager in die Lage versetzen soll, zeitnah Fragen zu beantworten wie: »Wie hoch war der Gesamterlös für das Regionalgebiet Ost im dritten Quartal?« (Forsman 1997).<sup>cxxxvii</sup> Neben diesen Analysen vergangener Ereignisse, sollte OLAP das unternehmerische Bedürfnis nach Simulation zukünftiger Entwicklungen bedienen. »Was wäre die Auswirkung auf die Kosten eines Softdrinckes, wenn die Sirup-Preise um 0,10 Dollar pro Gallone steigen würden und sich die Transportkosten um 0,05 pro Meile sanken?«<sup>cxxxviii</sup> (ebd.). Die für Simulationen des gleitenden Durchschnittswert oder des prozentualen Wachstum benötigten Formeln sollten bereits integriert sein, sodass die Nutzer sich nicht den Kopf über mathematische Details zerbrechen brauchten.

Auffallend ist, dass in OLAP-Anwendungen Zeit und Zeiträume das verbindende Element der Fragestellungen waren und sind. Es lag daher nahe, dem Verlauf von Zeit in den Interfaces und den bereitgestellten Modulen und Abfragen besondere Aufmerksamkeit zu widmen. Dieses informatorische Bedürfnis führte zu einer Wahrnehmung, die sich aus der Fläche der Tabelle verabschiedete und eine dritte Achse hinzufügte, die häufig der Darstellung einer zeitlichen Dimension diente. »Anstatt von Zeilen und Spalten, bekämen Nutzer ein Puzzle von Informationen, in welchem jeder multidimensionaler Block mit jedem anderen Block verknüpft ist. Während eine Spalte Informationen über Produktbezeichnungen, Verkaufserlöse und Ausgaben für eine Zeile mit Produkten (wie Nägel, Schrauben, Hämmer) enthielte, könnten OLAP-Werkzeuge Dimensionen wie Zeitabstände, Verkaufsgebiete und Distributionskanäle hinzufügen« (Codd/Edgar F./Codd/Salley 1993, 88).<sup>cxxxix</sup> Aus der Tabelle, welche

den Tupeln der mathematischen Menge ein Interface verleiht, wird nun ein Würfel, welcher die Tupel mit der Metapher einer zusätzlichen räumlichen Dimension in der Zeit verortet (Abb. 50).

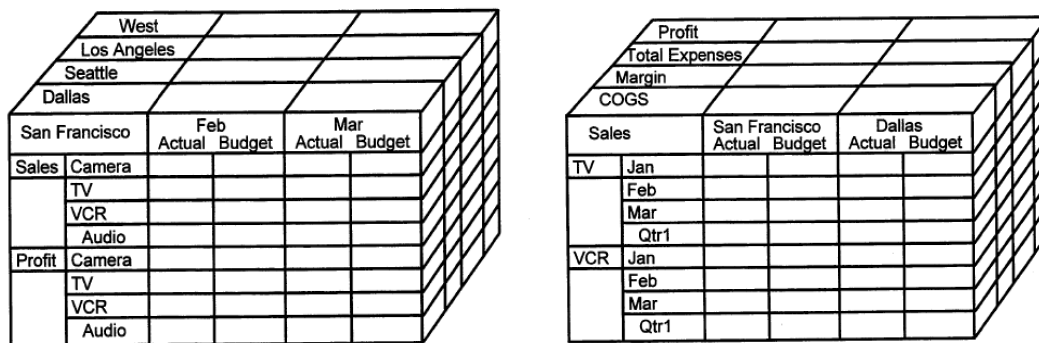


Abb. 50: Illustration von OLAP-Würfeln in der Patentschrift für Essbase von Arbor Software Corp. Drei Dimensionen eines fünfdimensionalen OLAP-Würfels. Links: Die Vorderseite zeigt die tatsächlichen Verkaufszahlen (Spalte: Actual) und Gewinn für verschiedene Produkte (TV, VCR, Audio) im Vergleich zu den geplanten Zahlen (Spalte: Budget) über einen Zeitraum von Februar und März für die Verkaufsregion San Francisco. Dahinter befinden sich die gleichen Übersichten für andere Regionen (Dallas, Seattle, Los Angeles, West). Rechts: Der gedrehte Würfel stellt drei andere aus fünf Dimensionen dar und erlaubt es, Verkäufe, Gebühren, Gewinnmarge, Gesamtausgaben und Gewinn für die San Francisco und Dallas und verschiedene Geräte (TV, VCR) über einen Zeitraum zu vergleichen, ebenfalls unterschieden nach Plan und tatsächlichen Verkaufszahlen (Earle 1994).

Die Proponenten der OLAP-Würfel entwickelten mit der Zeit einen eigenen Terminologie: Navigation in Daten und das Erzeugen von Übersichten wird in der Metapher des Würfels bleibend als Slice and Dice bezeichnet, weitere Operationen erhalten Würfel-Konnotationen wie Drill Down/Up, Roll-Up oder Pivot.

Edgar F. Codd, Sharon B. Codd and C.T. Salley, die gemeinsam mit Chris Date nach ihrem Weggang von IBM in der Consulting-Firma E.F. Codd Associates arbeiteten, veröffentlichten 1993 ein Whitepaper, in dem sie die Analyse von Datenmengen aus relationalen Datenbanksystemen erstmals als On-Line Analytical Processing bezeichneten (Codd/Edgar F./Codd/Salley 1993, 88). Der Begriff OLAP war erfolgreich in die Welt gesetzt und diente einer ganzen Reihe von Softwareprodukten zur Identifizierung, die vorher u.a. als Executive Information System (EIS), Grafik- und Reportgeneratoren oder Tabellenkalkulation bezeichnet wurden.<sup>198</sup> In dieser Zeit waren E.F. Codd Associates für eine Analyseprodukt namens Essbase tätig, welches als OLAP eingesetzt wurde. Essbase ging fünf Jahre später durch eine Firmenverschmelzung an die Hyperion Solutions Corp. über, welche wiederum 2007 durch Oracle Inc. aufgekauft wurde. Heute ist Essbase Teil der Oracle Analytics Cloud. Essbase wurde zudem auch mit DB/2 von IBM eingesetzt, benannt als DB2 OLAP Server (IBM 2001, xi).

IBMs offizielle Dokumentation zu IBM Essbase von 2001 erwähnt einleitend Peter Senge's *The Fifth Discipline – The Art and Practice of the Learning Organization* von 1990, eines der populärsten Management-Bücher seiner Zeit mit über einer Million verkauften Exemplaren. IBM Essbase wird darin als Anwendung gerahmt, welche in der Lage sei, die durch Senge identifizierten »dynamischen

<sup>198</sup> Die Breite von Analysesoftware ist überraschend. So zählen dazu die Programmiersprache APL (IBM, 1962), die Software Express (Information Resources, 1975; später von Oracle übernommen), System W (Comshare, 1982), Metaphor (Metaphor, 1984; später von IBM übernommen), Command Center (Pilot Executive Software), Commander (Comshare, 1984); (Pendse 2007). Spreadsheet Software wie Visicalc (1979), Lotus 1-2-3 (1983), und Microsoft Excel (1985) wurden in Größenordnungen eingesetzt und verdienten ein eigenes Kapitel (Siehe Campbell-Kelly In: Campbell-Kelly u. a. (Hrsg.) 2003, 323–348).

Prinzipien« des System-Denkens, »welches uns tatsächlich helfen kann, mit komplexen Geschäftsumgebungen umzugehen« (ebd.),<sup>cxl</sup> abzubilden.

Senge beschrieb eine Managementtheorie der lernenden Organisation, die auf mehreren Säulen basierte:

- Erstens würde die individuelle Aneignung eines Gegenstandes als kontinuierlicher, nicht-ender Lernprozess aufgefasst.
- Zweitens müsse klar sein, dass die eigenen mentalen Modelle auf Vorannahmen, Erinnerungen und Erfahrungen basieren und daher nicht zwangsläufig auf andere Mitglieder der Organisation anzuwenden seien.
- Drittens sollten strategische Überlegungen auf gemeinsam entwickelten Zukunftsvisionen basieren, die nicht oktroyiert werden sollen.
- Daraus folge viertens, dass Lernen im Team intensiver erfolge als individuell und die kollektive Intelligenz vorzuziehen sei. Die Teammitglieder wäre somit in der Lage, unvorhergesehene Lösungen für Probleme zu identifizieren (Senge 1990, 172ff., 190ff., 211ff., 257ff.).

Die Analyse von Unternehmensdaten mit OLAP-Anwendungen wie Essbase erlaubte in diesem Sinne eine In-Formatisierung neuer Anforderungen, die sich im Zuge von Flexibilisierung der Produktionsketten und Globalisierung der Warenströme ergaben. Gleichzeitig ermöglichten die Simulationsmöglichkeiten in OLAP und ähnlichen Anwendungen eine weitere Granularisierung dieser Warenströme in Bezug auf Absatzmärkte, Produktdifferenzierungen und Marketing. Das Systemdenken (à la Senge) zeigt Überschneidungen zur Kybernetik, vor allem wenn es um die Rekursivität des Lernprozesses geht. Eine Untersuchung, ob OLAP stärker im Feld der kybernetischen Operations Research zu verorten ist als im Feld der bürokratischen Informationsverarbeitung, bleibt an dieser Stelle Desiderat. Vermuten lässt sich hier eine Brückenfunktion bürokratischer und kybernetischer Genealogien.

Zusammenfassend ermöglichte OLAP vier Qualitäten:

- die Zusammenführung disparater Daten in einem Data Warehouse,
- multiperspektivische Ansichten von Informationsobjekten,
- die rechenintensive Aggregation von Daten und
- die Herstellung von zeitlichen Bezügen durch den Einsatz von Zeitreihen.

Wie ist das Auftreten von OLAP in soziale Veränderungsprozesse einzuordnen? Arthur P. Mols wertete in Anschluss an Manuel Castells die aktuellen Entwicklungen als Übergang von einer Informations-Ökonomie zu einer informationellen Ökonomie und Gesellschaft, befindet Manuala Ott. »Diese informationelle Gesellschaft unterscheidet sich von der alten Informationsgesellschaft darin, dass sie Informationstechnologien aufs engste mit sozialen Organisationsformen verflucht und die Generierung und Übertragung von Informationen zur fundamentalen Ressource ihrer Produktivität und Ermächtigungsstrategien erhebt. Denn ihre Produktivität und Wettbewerbsfähigkeit hängt von der Fähigkeit ihrer Akteur/innen ab, Information zu generieren, zu prozessieren, aber auch rückzukoppeln, zu kritisieren und umzuschreiben« (Ott 2015, 253). Die von Ott angesprochene Verflechtung der sozialen Organisationsformen mit Informationstechnologien geht einher mit registrierenden, delegierenden und koordinierenden Praxen, die sich von der doppelten Buchhaltung bis OLAP kulturtechnisch durch die Geschichte der Menschen und ihrer Apparate ziehen. Humane Akteure werden sich immer wieder auf's Neue gewahr, welchen epistemischen Zugewinn Abstrakta in Form von Daten bieten.

In der kapitalistischen Logik sind diese notwendigerweise warenförmig organisiert: Das im Zusammenhang mit OLAP etablierte Data Warehouse verweist in der sprachlichen Dimension auf eine Vorstellung von Daten, welche diesen Wert und Mehrwert zuschreibt, denn es waren Güter und Waren, für welche die Lagerhäuser entstanden. Relationale Datenbankmanagementsysteme erscheinen nicht mehr allein als Figuration, in der formatisierte Informationen strukturiert abgelegt und zur Verarbeitung abgerufen werden können, sondern auch als Ökonomisierungskatalysator von Daten.

### 2.3.8 Fazit

»Jede Transaktion generiert Informationen über sich selbst, welche dann eingesetzt werden, um weitere Entscheidungen der Konsumenten zu leiten« (Bauman/Lyon 2013, 104).<sup>cxli</sup>

Das Vorhaben dieses Kapitels war es, das maschinisierte und automatisierte »Delegieren« von Informationen, welches in relationalen Datenbanken als Transaktion gefasst ist, zu untersuchen. Dabei konnte die Transaktion als medial-ökonomische Praxis beschrieben werden, welche menschliche Akteure und Computer miteinander integriert und zwar im Modus der Arbeitsteilung. Es schälte sich dabei heraus, dass die datenbanktechnische Implementierung der delegierenden Transaktion eine Skalierung und Kaskadierung erlaubt, welche als übergeordnete Koordination erscheint. Transaktionen in Datenbankpraktiken sind, um mit Alex Galloway zu sprechen, Protokolle, welche verteilte und heterogene Management-Systeme der Kontrolle unterwerfen (Galloway 2004, 7f.).

Die Transaktion ist ein wichtiges Beispiel dafür, wie eine spezifische Medienpraxis in der Datenbanksoftware zu Technologie wird. Sie ist nicht irgendeine Medienpraxis, sondern eine der wichtigsten Medienpraxen, um die wechselseitige Koordination von Prozessen durch verschiedene Infrastrukturen hindurch zu garantieren und diese Garantie zu kontrollieren. Die Transaktion wurde durch die Informatik übersetzt, indem dem wirtschaftlichen Handeln zugrunde liegende Aktionen und Operationen dem Register des Maschinischen verfügbar gemacht wurden, und zwar als eine Kette von Operationen, die entweder vollständig oder gar nicht durchgeführt werden.

Was als eine Untersuchung begonnen hat, wie transaktionale Praxen zu einer spezifischen Datenbanktechnologie wurden, ist nun noch einmal neu zu fassen. Ging es ursprünglich um die Frage der Transaktionsdaten, so stehen am Ende dieses Abschnitts die aus Daten gewonnenen Analysen im Vordergrund, denn durch Verwendung von Transaktionsdaten tritt in der Gegenwart der *informational economy* und *society* in eine neue Qualität sozialer und ökonomischer Verhältnisse ein.<sup>199</sup> In einer Reihe von Schritten soll dieser Übergang markiert werden:

Erstens, »Prozesswissen«: Transaktionen in DBMS generieren ein erweitertes Prozess-Wissen, welches mit den vorherigen Methoden der Wissensorganisation nicht vorhanden war. Dies betrifft einerseits Kategorien der Zeit und andererseits des Raums, die wesentlich feiner erfasst werden können. Aktuelle Beispiele dafür sind Daten zur Verkehrsdichte, die anhand der Bewegungsdaten von Smartphones registriert werden (Stenovéc 2015), oder die Auflösung geographischer Kategorien: Orientierte sich staatliche Statistik bisher an den Gebietsgrenzen der Behörden, die die Daten erhoben, so können nun

---

<sup>199</sup> Siehe hierzu auch *Datenarbeit im »Capture«-Kapitalismus. Zur Ausweitung der Verwertungszone im Zeitalter informatischer Überwachung* (Heilmann 2015, 39).

Nutzerprofile Einblicke in geographische Gebiete geben, die derartige Grenzziehungen überschreiten (vgl. Rudder 2014, 199). Immer basiert dieses Prozesswissen auf Transaktionen und den Analysedaten welche aus diesen gewonnen werden.

Zweitens, ›Optimierung mittels ökonomischen Wissens‹: Transaktionsdaten und die darauf basierenden Analysedaten werden zur betriebsinternen, technischen und prozessorientierten Geschäftsoptimierung eingesetzt: zur Berechnung von Geschwindigkeiten, Produktivität der Mitarbeiter, Kosten der Betriebsabwicklung, zum Beispiel Lagerkosten, Retourkosten und so weiter. Sie verflochten sich mit allgemeinen betriebswirtschaftlichen Analysen. Ab dem Moment, ab dem ein auf Transaktionsdaten basierender Wissensvorteil als vorteilsgebend gegenüber der Konkurrenz erkannt wurde, ist es unter kapitalistischen Bedingungen unausweichlich, diesen einzusetzen. Dies formt, wie im Kapitalismus über Transaktionsdaten und deren Analyse gedacht wird und überhaupt gedacht werden kann. Voraussetzung für die Ökonomisierung von Daten war es, Aktionen als Transaktionen zu fassen und in Datenbanken zu registrieren. Dieser medial-ökonomische Prozess kann inzwischen als historisch größtenteils ausformuliert gelten.

Drittens, ›Datenbank als eine stabil-instabile Figuration‹: Technologischer Knotenpunkt für die logistischen Delegationen ist die relationale Datenbankfiguration. Medientheoretisch verschiebt sich die Beschreibung elektronischer Datenbanken als stabiles oder stabilisierendes System: Indexe und laufzeit-kompilierte Optimizer, die auf unterschiedliche Abfragen flexibel reagieren, erzeugen eine ›stabil-instabile Figuration‹ von Algorithmen, Daten und Informationsmodellen, welche ständigem Wandel unterliegen. Die Notwendigkeit verschiedener gleichzeitiger Nutzerzugriffe und atomischer Transaktionsabwicklungen führt in einen Dauerzustand der Inkonsistenz, der allein an bestimmten Checkpoints einem kurzen Zustand der Konsistenz weicht – immer dann, wenn alle Transaktionen abgeschlossen und in den Speicher geschrieben sind. Die relationale Datenbank ist von Checkpoint zu Checkpoint instabil, also den größten Teil ihrer Verwendung über. Transaktionen stellen nun Einheiten von Konsistenz dar – nicht länger bilden sie allein handelnde Realität ab. Die relationale, instabile Datenbank wird so zum Werkzeug, welches sich in einem beständigen Übergang von einem zum nächsten konsistenten Zustand befindet.

Viertens, ›Materialität von Daten‹: Die In-Formatisierung der Transaktion entsprang einer Reihe von Transformationen und zeichnete sich historisch aus durch nicht-linearen Speicherzugriff, entwickelt aus der Beobachtung buchhalterischer Abläufe und durch die Maschinenunabhängigkeit von Daten, durch die Priorisierung von Vorgängen für die Prozessierung innerhalb eines Mainframe-Computers. Die Transaktion zeichnet sich aus durch einen Zugang, der die Daten als Repräsentation von Aktionen zentral stellt und Programme und deren Zugriffe als Anfragen auf Daten fasst, sowie durch den interaktiven On-Line-Zugriff, der für eine geographische Distribution sorgt. Das verschob den Begriff der Transaktion, der nun nicht nur ein ökonomischer und juridischer, sondern auch ein technologischer Begriff wurde. In Kontrollsphären formalisiert, etablierte sich im relationalen Datenbankmanagementsystem eine Mikrohistorizität, die es erlaubte, einzelne Transaktionen teils oder ganz rückabzuwickeln.

Vor dem Hintergrund dieser Entwicklungen ist schließlich eine Unterscheidung einzuziehen, welche auf die ›Materialität von Daten abzielt:

- stabile Stammdaten,
- dynamische Transaktionsdaten
- Analysedaten
- Nicht-Daten

Dies bietet die Grundlage für ein weiteres medientheoretisches Nachdenken über die Bedeutung von Daten in heutigen Anwendungen.<sup>200</sup>

Fünftens, ›Permanente Aktualisierung des Gegenstandsstatus‹: Durch Transaktionen werden die Gegenstände eines Prozesses beständig aktualisiert. Dies gilt für das Konsumenten-Waren-Verhältnis ebenso wie für das Material-Produktionsverhältnis.

Zum einen ändert sich der Status von Konsumenten und von Waren, seitdem Transaktionen oder die Muster, welche Transaktionen erzeugen, beständig an einzelne Personen rückgebunden werden können. Diese permanente Ansprache (im Sinne von Althusser's Apellation) mittels Smartphone erfolgt durch Verwendung eines registrierenden Logins. Der Kauf einer Ware war ursprünglich in der doppelten Buchhaltung zwar belegt durch Kassenbon, Quittung oder Rechnung. Grundsätzlich jedoch konnten diese Transaktionen als zeitlich und räumlich abgeschlossen gelten, sobald der Tausch vollendet war. Erst mit der Einführung der Kreditkarte werden Konsumenten ständig und jederzeit identifizierbar und erst seit den 1970er Jahren werden getätigte Bankentransaktionen lückenlos in Datenbanken nachweisbar. Seitdem sind die Käufer von Waren dauerhaft durch Transaktionsdaten bei den Herstellern registriert und durch diese permanent ansprechbar.

Zum anderen gilt ähnliches für Objekte, welche mit Hilfe transaktionierender Datenbanken verzeichnet und verfolgt werden. Seien es die Bauteile eines Motors, die in den Stücklisten verwaltet werden, oder die Pakete, deren postalischer Weg beständig verfolgbar ist: Die in Transaktionen verwickelten Gegenstände von Prozessen werden nie ›in Ruhe‹ gelassen. Immerzu wird ihre Status aktualisiert und ist auf Abfrage verfügbar und zwar im Falle der Produktregistration über den Verkaufsakt hinaus. Des weiteren suchen Unternehmen nun zunehmend Möglichkeiten, statt einzelner Warenverkäufe, regelmäßig wiederholende Dienste anzubieten, die einen kontinuierlichen Umsatz generieren. Signifikantes Beispiel ist die Umwandlung von Kaufsoftware in Mietsoftware<sup>201</sup> oder genereller gesagt das Konzept der Lizenzierung digitaler Güter.<sup>202</sup> Menschen und Objekte werden einer neuen Form der partiellen Repräsentation zugeführt, welche nicht nur ihre Stammdaten, sondern auch die damit verbundenen Transaktionsdaten und Analysen als Databody (Mattern 2018) in Datenbanken ablegt.

Sechstens, ›Veränderungen im Feld des Privaten‹: In der Konsequenz sind transaktionierte Waren und Dienstleistungen nicht mehr dem Feld des Privaten, im Sinne klassischen Privateigentums,

---

<sup>200</sup> Eine Bewertung, wie sich diese Materialitäten von Daten zum Konzept der Metadaten verhalten, steht aus und muss andernorts geleistet werden. Zum Konzept der Metadaten siehe beispielsweise *Understanding Metadata* (National Information Standards Organization 2004). Metadaten sind Daten über Daten, sie beziehen sich auf den Zweck einer Sache.

<sup>201</sup> Zum Beispiel der Übergang von der Software Adobe Creative Suite zum Abo-Dienst Adobe Creative Cloud ab 2013.

<sup>202</sup> Hier ließe sich hinterfragen, ob nicht gerade die Open Source und Creative Commons Lizenzen eine Vorreiterrolle gespielt haben im Bewusstwerdungsprozess über die Profitabilität von Lizenzmodellen als unabgeschlossenen Vorgang. Da sich Open Source Software nicht unmittelbar verkaufen ließ, wichen ihre Anbieter auf Servicemodelle mit wiederkehrenden Gebühren aus, die Installation, Wartung und Updates betrafen.



zuzuschlagen. Zumindest für transaktionierte Waren/Services lässt sich feststellen, dass die Konsumenten über diese lediglich in Form von Lizenzen verfügen, sie besitzen sie nicht mehr dauerhaft, abgeschlossen und unveränderbar. Aus Konsumenten wurden Nutzer. Sie sind durch Serviceangebote beständig verbunden, generieren immer neue Transaktionsdaten, wo früher Stille herrschte. Wir erleben einen doppelten Moment der Transgression: Statistik und Individualität verschränken sich und die gesellschaftliche Stellung des Privateigentums verändert sich dauerhaft.

Siebtens, ›Datenzentrierte Wahrnehmung der Realität‹: Im Zuge der Datenanhäufung im Data-warehouse ab den 1990er Jahren und von Big Data ab Mitte der 2000er Jahre eskaliert die Produktion von Analysedaten, welche sich aus dem wirtschaftlichen Feld in das soziale Feld ausdehnt. Das Prinzip wird übergriffig, oder wie es Engemann formuliert: »kein Ereignis, keine Handlung, keine Kommunikation – oder Nichtkommunikation – die nicht potentiell transaktionsrelevant wären.« (Engemann 2014, 377). Wenn jegliche Kommunikation und Aktion mit der Erhebung von Transaktionsdaten einhergeht, wird sie einem unumkehrbaren Zwang unterworfen. Die ökonomische Notwendigkeit transaktionsbasierter Unternehmen erzeugt eine datenzentrierte Wahrnehmung der Realität, die sich daran orientiert, wie immer neue Transaktionsdaten zu generieren sind, wie beispielhaft an Social-Media-Unternehmen und den darum gruppierten Werbepattformen sichtbar wird. Transaktionsanalysen entstehen nicht länger als Beiprodukt menschlicher Aktivitäten, sondern werden zu deren Ausgangspunkt. Das informatorische Potenzial tritt den Menschen als Imperativ gegenüber.<sup>203</sup> Wenn das eine Unternehmen OLAP einsetzt, um in Verschränkung mit den Managementtechniken lernender Organisationen die eigene Konkurrenzsituation zu verbessern, so stehen auch die anderen Unternehmen im Zugzwang, ein entsprechendes Analyseregime über Transaktionsdaten zu installieren. Aus der Analysemöglichkeit wird Analysezwang. Permanente Datenerhebung wird zum Normativ.

Achtens, ›Ausweitung der Transaktionszone‹ (Engemann 2014, 377): In diesem unbewussten Prozess, der sich hinter dem Rücken der Menschen vollzog, entfaltete sich der Übergang von der *information economy* zur *informational economy and society*. Er setzt sich heute darin fort, dass die Transaktionalisierung des menschlichen Handelns auf das Feld sozialen Handelns und sozialer Kommunikation ausgeweitet wird. Ermöglicht wird damit eine massenhaft teilindividualisierte Appellation von Individuen und eine Ausweitung der Wertschöpfung auf den bisher davon weitestgehend ausgenommenen Bereich der sozialen Interaktion.

---

<sup>203</sup> Es bleibt als Desiderat, anderen Nutzungen von Transaktionsdaten in der OLAP-Methodologie, z.B. in Kunst und Kultur, sowie Wissenschaften nachzugehen. Damit ließe sich eine hier möglicherweise zu starke Betonung des ökonomischen Feldes relativieren. Andererseits sind auch diese Felder neo-liberalen Strategien der In-Wertsetzung unterworfen.

## 3. Technoemergenz im Ost/West-Vergleich

### 3.1 Kontexte der Fallstudien

»Wie selbstverständlich konzentrierte die DDR dabei ihren ganzen Ehrgeiz auf die Hardware und auf deren Kernstück: die Chips, nicht auf die Software, die Programme. Das entsprach der traditionellen deutschen Ingenieur-Mentalität; auch bei Siemens dachte man anfangs ähnlich« (Radkau 1990, 20).

Die Historiographie von Software, der damit verbundenen Infrastrukturen und Kulturtechniken ist dann am reichsten, wenn es gelingt, nicht linearen Fortschritt, sondern Seitenpfade, Unterbrechungen und Abweichungen zu untersuchen. Zwei Fallbeispiele des Umgangs mit elektronischen Datenbanken und der Einführung des relationalen Modells in der DDR werden daher in diesem Abschnitt vor der gebrochen-teleologischen Folie der Technologiesgeschichte der DDR detailliert analysiert.

Die Fallbeispiele überschneiden sich grob mit den durch Gerhard Merkel 2006 vorgeschlagenen Periodisierung der Rechentechnik in der DDR. Die durch Merkel markierte Pionierphase »1948–1960: Pioniere können ihre Ideen verwirklichen« (Merkel 2006, 50–53) wird übersprungen, da sich die Pionierleistungen zuvorderst um die Einführung elektronischer Rechentechnik und später auch der zugehörigen Betriebssysteme, handelte. Erst mit dem Vorhandensein dieser technologischen Voraussetzungen und der neuen Informations-Bedürfnisse, welche diese »Pionier«-Figurationen ermöglichten, beginnt eine zunehmende Verkomplizierung der Datenhaltung und die Untersuchung von Datenbankmanagementsystemen als Lösungsansatz.

Als Vorspann dient das Kapitel 3.1 Kontexte der Fallstudien, welches der Orientierung der Lesenden dient, und Institutionen der Planung und Lenkung in der DDR, den Aufbau des Kombinat Robotron, das Embargo des Coordinating Committee on Multilateral Export Controls (COCOM), die Praxis der Software Adaptionen, vorstellt, und eine einführende zeitliche und topologische Übersicht über Datenbanksysteme in der DDR gibt.

Das Kapitel 3.2 *Robotron Dresden: Sachgebietsorientierten Programmiersysteme (SOPS) – Das »Sozialistische SAP«?* überschneidet sich zeitlich mit der durch Merkel markierten zweiten Periode »1958–1971: Starke Förderung von EDV und Automatisierung« (ebd.) und untersucht die Emergenz von Datenbankmanagementsystemen im Rahmen der ebenfalls emergenten Ressourcenplanung von Produktionsmitteln, welche in Management Resource Planning (MRP), Enterprise Resource Planning (ERP), Supply Chain Management (SCM) und Customer Relationship Management (CRM) mündete. Hierfür gelang es in der DDR einen innovativen modularen Korpus an Software zu schaffen, welche Aufschluss über die Bedürfnisse einer hochindustrialisierten Gesellschaftsformation gibt.

In die durch Merkel postulierte dritten Periode »1971–1983: Stagnation in Entwicklung und Anwendung der Rechentechnik« (ebd.) fällt die Eigenentwicklung eines relationalen Datenbankmanagementsystems in Zusammenarbeit der Technischen Universität Dresden und des Kombinates Robotron. Welchen Schwierigkeiten und Wendungen diese Entwicklung unterlag, aber auch wie sie anhand limitierter Ressourcen ermöglicht wurde, diskutiert das Kapitel 3.3 *Kalifornien, Technische Universität Dresden und Robotron Dresden: Das Problemorientierte Seminar »Datenbanken« und DABA 1600*.

Es handelt sich bei den Fallbeispielen um keine reine Geschichtsschreibung, sondern um Genealogien, welche an bestimmten Punkten einsetzen und in nicht notwendigerweise linear dargestellt werden. Vielmehr orientieren sich die Kapitel am untersuchten Material und oft springen sie zu westlichen Vergleichsbeispielen, welche einer Gesamtverdichtung des Untersuchungsgegenstandes dienen sollen. Den Lesern ist in diesen Sprüngen einige Geduld abverlangt, jedoch verspricht die Reichhaltigkeit des Materials eine epistemische Dichte, welche sich nur durch diese Überlagerungen und Durchdringungen der zeitlichen und topologischen Räume verwirklichen ließ. Damit soll das Versprechen eingelöst werden, die heutige Durchdringung des Alltages mit relationalen Datenpraktiken durch das Herausarbeiten von Emergenzen und Variantologien erklärbar zu machen. Bevor die Fallbeispiele erörtert werden können, sind dafür einige kontextuelle Voraussetzungen zu schaffen.

Die folgenden Abschnitte konzentrieren sich mit der DDR auf eine gesellschaftliche Konstellation und einen geographischen Fokus, welcher sich selbst als ›Realsozialismus‹ verstanden hat. Um dieser Spezifik gerecht zu werden, werden vorab einige Kontexte aufgerufen, die auch wenn sie zum Teil bekannt sein könnten, hier in aktualisierter Form und zugespitzt auf den Untersuchungsgegenstand beschrieben werden.

Da vielfach Institutionen der Planung und Lenkung in der DDR angesprochen werden, stellt ein erster Abschnitt dar, wie diese in Bezug auf die Entscheidungsfindung und Durchsetzung von Beschlüssen angelegt waren. Dabei werden Unterbrechungen sichtbar, welche die Idee, dass in der DDR durch die Sozialistische Einheitspartei Deutschlands (SED) von Oben nach Unten durchregiert wurde, zugunsten einer komplexeren Sichtweise modifizieren. Anhand zweier Fallbeispiele werden die Aushandlungswege, die Widerstände und Entscheidungsnetze sichtbar gemacht, welche die Entwicklung von Mikroelektronik und Software in der DDR prägten. Ein weiterer Kontext ist die geraffte Betriebsgeschichte des Kombines Robotron, die in Form eines Überblicks über dessen komplexe Zusammensetzung aus zahlreichen Einzelbetrieben, gegeben wird. Robotron war in der DDR der zentrale Hersteller von Rechentechnik und Software, eine Infrastruktur, welche basierend auf den verschiedenen Beschlüssen des Staates DDR und der Partei SED geschaffen wurde. Diese Übersicht dient nicht nur der geographischen und institutionellen Verortung, sondern auch einem Verständnis, in welchen Abteilungen an Software und Datenbanksoftware gearbeitet wurde, und wie dies in den Gesamtkomplex Robotron eingebunden war. Ein in den Interviews und der Sekundärliteratur mehrfach auftretender Topos sind die westlichen Exportbeschränkungen durch das Embargo des Coordinating Committee on Multilateral Export Controls (COCOM). Dabei wird COCOM häufig als ein bedeutender Grund für Eigenentwicklungen und Adaptionen von Hard- und Software genannt. Ein weiterer aufzurufender Kontext ist die Praxis der Software-Adaptionen, in deren Rahmen westliche Software für DDR-Computer angeeignet und ›adaptiert‹ wurde. Anstatt jedoch diese Adaptionen schlichtweg als Appropriation zu verstehen, sollen diese als spezifische Form des Wissenstransfers untersucht werden, welcher sich in Softwaretechniken wie Reverse Engineering, Refactoring, Forking, Debugging und Disassemblierung einreicht. Schließlich erfolgt eine Orientierung über die in der DDR verfügbaren Datenbankmanagementsysteme. Diese zeitliche und topologische Übersicht begegnet dem Umstand, dass die ausführlicher ausgearbeiteten Fallbeispiele notwendigerweise eine ganze Reihe anderer Datenbankentwicklungen nur unzulänglich darstellen können.

Diese kontextuellen Setzungen beabsichtigen, eine Rahmung für das folgende Textverständnis zu geben. Sie geben die nötige Ortskenntnis für die Fallbeispiele der innovativen Sachgebietsorientierten Programmiersysteme und der relationalen DDR-Eigenentwicklung DABA 1600.

### 3.1.1 Institutionen der Planung und Lenkung in der DDR

»Im Zeitraum 1971 bis 1975 wurde gegenüber einer vorgesehenen Auslastung [der vorhandenen Rechentechnik – F.H.] von durchschnittlich 13 Std. je Kalendertag eine Größenordnung von rd. 15 Std. erreicht. Der Planung für den Zeitraum bis 1980 wurde eine solche Nutzung zugrunde gelegt, welche von den fortgeschrittensten Erfahrungen des abgelaufenen Fünfjahrplanes ausgeht«, heißt es im Editorial der Monatszeitschrift *rechentechnik/datenverarbeitung*, verfasst durch den Abteilungsleiter »Rechentechnik« der Staatlichen Plankommission (Pieper/Fritsch 1976, 1). Das Zitat ist in mehrfacher Hinsicht prägnant. Es ist ein Beispiel für einen von mehreren Kommunikationskanälen für die Planziele an diejenigen, die sie umsetzen sollten, die Facharbeiter für Datenverarbeitung, Betriebsdirektoren, SED- und Gewerkschaftskader. Es ist auch ein Beispiel dafür, wie sich die Planung an Quantitäten orientierte und für die fortwährenden Versuche, die Quantitäten zu steigern.

Damit steht die Frage im Raum, wie diese Planvorgaben, die gesetzlichen Rahmenbedingungen der Wirtschaftsentwicklung, und damit der Automatisierten Datenverarbeitung, und die Beschlüsse zur grundsätzlichen Ausrichtung der DDR-Gesellschaft ausgehandelt wurden. Im Anschluss an eine Überblicksdarstellung der Institutionen der DDR, sollen zwei knappe Beispiele den Aushandlungsprozess beleuchten.

Zuerst sollen die Planung aus Perspektive der maßgeblichen Partei gezeigt werden: Die Parteitage der Sozialistischen Einheitspartei Deutschlands (SED) gaben die wirtschaftspolitische Grundausrichtung vor. Beschlüsse der Parteitage dienten als grundlegendes Argumentationsmaterial für die Durchsetzung von Gesetzgebung oder auch von Beschlüssen des Zentralkomitees der SED, des Ministerrats der DDR oder der Staatlichen Plankommission. Damit war in der DDR die Vormachtstellung der Partei SED gegenüber dem Staatsapparat gesichert: Entscheidungen wurden während der Parteitage öffentlich inszeniert und im Zentralkomitee der SED gefällt (Wilke/Schroeder 1996a). Das Zentralkomitee der SED bestand aus 100 (1950) bis 215 (1986) Teilnehmenden, und traf mehrmals im Jahr zu Sitzungen zusammen. Die Teilnehmenden wurden durch die Parteiführung vorgeschlagen und auf den Parteitag bestätigt (Wilke/Schroeder 1996c). Das Politbüro des Zentralkomitees der SED, als innerer Zirkel, mit 15–25 Mitgliedern traf sich im wöchentlichen Abstand. Dazu gehörten 12 Sekretäre des Zentralkomitees, welche gleichzeitig Leiter der thematischen Arbeitsgruppen waren, und die inhaltliche Arbeit betrieben (Wilke/Schroeder 1996b; vgl. Lepsius 1995, 347–354).

Nun zu den staatlichen Organen: Der Ministerrat war nominal die Regierung DDR. Die an den Ministerrat angegliederte Plankommission arbeitete den Ministerien und dem Zentralkomitee der SED zu, indem sie Wirtschaftsdaten sammelte, verdichtete und die Wirtschaftspläne für die Fünfjahresplanung erstellte. Zur Erstellung des jeweils nächsten Wirtschaftsplanes (mittelfristige und jährliche Planung) verhandelte die Plankommission mit den Betrieben und Institutionen die zu erbringenden Leistungen aus, und setzte die aus dem Zentralkomitee der SED geforderten Leistungssteigerungen soweit möglich durch (Gutmann 1996; vgl. Lepsius 1995, 354–357). Die Ministerien waren in diesem Gefüge für die Umsetzung der durch das Zentralkomitee der SED getroffenen Beschlüsse zuständig. Obwohl formal der Ministerrat der DDR die Regierung darstellte, war er de facto den Parteigremien der SED nachgeordnet.

Wie verliefen die Entscheidungsprozesse? Für Beschlüsse und Gesetze wurden im Zentralkomitee der SED Arbeitsgruppen<sup>204</sup> gebildet, denen zahlreiche Mitarbeiter zugeordnet waren, und in denen neben

---

<sup>204</sup> Gegliedert in die Sachthemen: Außenpolitik, Agitation, Frauen, Kader, Internationales, Jugend, Kultur, Gesellschaftswissenschaftliche Institute beim ZK der SED, Sicherheit, Wirtschaft, Zahlungsbilanz.

den fachlichen Vertretern des Zentralkomitees der SED, Vertreter der Ministerien ihre Meinung äußern konnten und weitere sachkundige Entscheidungsträger, wie beispielsweise Betriebsdirektoren oder Fachleute vertreten waren. Die Leiter der Arbeitsgruppen waren als Sekretäre des Zentralkomitees zuständig für die Erstellung von Beschlussvorlagen und das Einholen zahlreicher Stellungnahmen, bevor Gesetzesentwürfe oder Beschlüsse in das Politbüro und das Zentralkomitee zur Abstimmung eingereicht werden konnten. Eine Beschlussfassung im Zentralkomitee und/oder im Politbüro war dann häufig Formsache. Den Betroffenen (z.B. Betriebsdirektoren) musste es gelingen, über die Arbeitsgruppen frühzeitig ihre Zustimmung oder abweichende Meinung in der Findungsphase kund zu tun und durchzusetzen, da in den Abstimmungen selbst nicht mehr kontrovers diskutiert wurde, sondern hauptsächlich die Entscheidungen angenommen wurden. Dem Generalsekretär des Zentralkomitees der SED (Walter Ulbricht bis 1971, danach Erich Honecker) stand es in diesem Prozess zu, Veto einlegen (Wilke/Schroeder 1996c).<sup>205</sup>

Die Aushandlungswege, die Widerstände und Entscheidungsnetze sollen exemplarisch in zwei Beispielen angerissen werden, im Bewusstsein darum, dass um deren Komplexität gerecht zu werden, eine eigene Untersuchung notwendig wäre.<sup>206</sup>

Das erste Beispiel illustriert die Notwendigkeit informeller und formeller Allianzen: Der Ingenieur für Regelungstechnik Gerhard Merkel war in seiner Funktion als Direktor für Wissenschaft und Technik des Zentralinstitutes für Automatisierung (ZIA) in Dresden, 1964 in die neu gebildete Regierungskommission für Rechentechnik berufen worden und stieg, obgleich ohne SED-Parteimitglied zu sein, 1966 zum Stellvertreter des Ministers für Elektrotechnik und Elektronik auf. Aus dieser Position heraus betrieb er im Rahmen der Regierungskommission die Gründung des Großforschungszentrums Robotron. Merkel konnte in seiner Funktion Vorschläge für die Plankommission und den Ministerrat unterbreiten, jedoch keine Entscheidungen treffen. Diese konnte sein Vorgesetzter der Minister für Elektrotechnik und Elektronik Otfried Steger bewirken. Ein weiterer Bekannter Merkels, sie hatten zusammen studiert, war Günther Kleiber. Dieser folgte ihm als stellvertretender Minister für Elektrotechnik und Elektronik und wurde außerdem Staatssekretär für Datenverarbeitung beim Ministerrat und später auch Kandidat des Politbüros der SED, sodass Merkel solange Kleiber in dieser Position war, von dieser Seite Unterstützung in seinen Anliegen erhielt. Der Vorsitzende der Plankommission Herman Große, der mit Merkel ebenfalls bekannt war, konnte Anliegen in die Plankommission zur Entscheidung einbringen.

Drei Arbeitsgruppen von Betriebsleitern, davon eine geleitet durch Merkel, unterbreiteten Ende der 1960er Jahre den Vorschlag für die Gründung des Kombinats Robotron. Als Begründungszusammenhang konnten sie auf das *Programm von Maßnahmen zur Entwicklung, Einführung und Durchsetzung der maschinellen Datenverarbeitung in der DDR in den Jahren 1964 bis 1970* verweisen, welches vom Politbüro des ZK der SED und vom Ministerrat der DDR 1964 verabschiedet worden war (Merkel

---

<sup>205</sup> Eine aktuelle und differenzierte Darstellung der Kommunikations- und Entscheidungskulturen im Zentralkomitee der SED leistet der Historiker Rüdiger Bergien mit der 2017 publizierten Studie *Im »Generalstab der Partei« – Organisationskultur und Herrschaftspraxis in der SED-Zentrale (1946-1989)*, (Bergien 2017). Zur Rolle der ZK-Abteilungsleiter vermerkte er: »Die wichtigste Machtressource [...] war daneben ihr Einfluss auf die Kaderpolitik in ihren Politikfeldern. Dieser Einfluss reichte so weit, dass sie sich »ihre« Minister selbst »backen« und darüber hinaus Generaldirektoren und Staatssekretäre, Professoren und NVA-Generäle in ihre Ämter bringen konnten« (ebd., 528).

<sup>206</sup> Hier ist auf die teils wertende Studie *Der schwierige Weg von der traditionellen Büromaschine zum Computer* von Erich Sobeslavsky zu verweisen, der die Etablierung der Hochtechnologie Mikroelektronik-Produktion in der DDR untersuchte (Sobeslavsky 1996).

2018, Fol 5–9; Ministerrat der DDR/Politbüro des ZK der SED 1964).<sup>207</sup> Von der Erkenntnis ausgehend, dass dieses Programm nicht im erforderlichen Maße umgesetzt wurde, formulierten sie eine Vorlage mit dem Ziel die Forschung, Entwicklung und Produktion der verschiedenen benötigten Teiltechnologien zu bündeln. Merkel war, wenn auch nach eigener Aussage distanziert, mit dem zuständigen Politbüro-Mitglied und Sekretär des Zentralkomitees der SED für Wirtschaft Günther Mittag bekannt. Mit diesem kam es u.a. im Zuge der von Merkel erarbeiteten Vorlage zum Kombinat Robotron zum Konflikt in Detailfragen, welchen Merkel argumentativ im Vier-Augen-Gespräch lösen konnte (ebd., Fol 8).<sup>208</sup>

Das hier skizzierte Beziehungsgeflecht lässt erahnen, wie grundlegend sowohl formelle als auch informelle Allianzen waren, um ein Vorhaben dieser Größenordnung – letztlich ging es um die Etablierung eines neuen Industriezweiges in der DDR – an den relevanten hierarchischen Knotenpunkten des DDR Partei- und Staatsapparates zu etablieren.<sup>209</sup>

Ein zweites Beispiel ist der *Beschluss des Ministerrates zur weiteren Entwicklung und multivalenten Nutzung von Software*, 1985. Darin sollte es um die Mehrfachnutzung von Software zur Ressourcenschonung gehen. Verantwortlich für die Redaktion war der Minister für Wissenschaft und Technik und stellvertretende Vorsitzende des Ministerrates, sowie Mitglied des ZK der SED, Herbert Weiz, welcher auch die entsprechende Arbeitsgruppe leitete. Der Verteiler für die Endredaktion ging als vertrauliche Dienstsache an die Mitglieder des Ministerrates, das Amt für Standardisierung, Messwesen und Warenprüfung, die Akademie der Wissenschaften der DDR, den Staatssekretär für Berufsbildung, das Amt für Erfindungs- und Patentwesen, die Staatliche Zentralverwaltung für Statistik, die Vorsitzenden der Räte der Bezirke, die Generaldirektoren der Kombinate, und die Kammer der Technik. Im Beschluss war u.a. festgelegt, welches Kombinat für welche Teilgebiete der Softwareproduktion zuständig war. Dies betraf neben dem VEB Kombinat Robotron elf weitere Betriebe und Institutionen (Ministerrat der DDR 1985, Fol 148–150 Anhang). Im Entwurf finden sich verschiedene Streichungen, für die allerdings heute die Entscheidungswege nicht im Detail nachvollziehbar sind. Beispielsweise wurde folgender Absatz gestrichen: »Vorhandene Software ist zu rationalisieren und an neue Anforderungsbedingungen anzupassen. Es ist ein effektives Verhältnis von Neuentwicklung und längerfristiger Nutzung vorhandener Software zu erreichen« (ebd. Fol 167 Anhang). In der Korrespondenz ist des Weiteren ein Schreiben des Instituts für Leichtbau und ökonomische Verwendung von Werkstoffen enthalten, welches die eigene Softwareproduktion, darunter automatisierte Planungssysteme, CAD-Software und ein

---

<sup>207</sup> Darauf basierend ist der Gang der Vorlagen beim ZK der SED, Abteilung Forschung und Entwicklung dokumentiert als: *Konzeption und Grundsätze der Organisation und Leitung für die Großforschung, Konzeption zur Einordnung von Wissenschaft und Technik die Leitung von Reproduktionsprozessen, Vorbereitung von Programmen für Großforschungsvorhaben, und Grundmodell eines Großforschungszentrums in Kombinat* (ZK der SED, Abteilung Forschung und Entwicklung 1968). Diese Konzeptionen führten zu mehreren Vorlagen im Ministerium für Elektrotechnik und Elektronik für ein Arbeitsgespräch am 24. März 1969 bei dem es konkret um die Gründung des Kombinat Robotron ging (Minister für Elektrotechnik und Elektronik 1969a). Einen Monat später, am 28.4.1969 fand ebenfalls im Ministerium eine Arbeitsbesprechung statt, die durch die *Vorlage der Bestätigungsdokumentation zur Herbeiführung der Grundsatzentscheidung für das volkswirtschaftlich strukturbestimmende Investitionsvorhaben »Komplexe Erweiterung des VEB Kombinat Robotron«* dokumentiert ist (Minister für Elektrotechnik und Elektronik 1969b). Der Arbeitsprozess führte schließlich zum *Beschluß betr. Grundsatzentscheidung über das volkswirtschaftlich strukturbestimmende Investitionsvorhaben »Komplexe Erweiterung des VEB Kombinat ROBOTRON«*, verabschiedet auf der Sitzung des Ministerrates der DDR am 23. Juli 1969 (Ministerrat der DDR 1969).

<sup>208</sup> Zur Nutzung des R300 im ZK der SED siehe *Im »Generalstab der Partei«* (Bergien 2017, 239–243).

<sup>209</sup> Vgl. auch für die SED-Zentrale *Mündliche Kommunikation* (Bergien 2017, 185–196) und für die sowjetische Planwirtschaft *Vertical Bargaining and other Organizational Dissonance in the Soviet Command Economy* (Peters 2017, 72–79).

Informationssystem für Materialwirtschaft berücksichtigt wissen wollte (ebd. Fol 185). Dieses Ansinnen wurde positiv entschieden. Für andere Vorschläge gibt es Stellungnahmen, warum ein Vorschlag abgelehnt wurde. Der Staatssekretär für den Bereich Investitionen und Wissenschaft/Technik der Staatlichen Plankommission Wolfgang Greß, übersandte kleinere Änderungsvorschläge, welche laut Schreiben »bereits auf dem Arbeitswege« kommuniziert waren. Dies verweist auf die sehr übliche Praxis, strittige Fragen im persönlichen Gespräch informell zu klären, bevor sie auf dem offiziellen Weg formuliert wurden (ebd. Fol 188f.). Am interessantesten ist eine Intervention durch die Akademie der Wissenschaften der DDR (AdW). Diese Institution schlug vor, für die im Gesetz vorgesehenen Softwareverzeichnisse, eine Datenbanksoftware der AdW zu verwenden (ebd. Fol 226f.). Dieser Vorschlag wurde abgelehnt. Für die Softwaredatenbank kam das Dokumentenrecherchesystem SIBE auf Basis von AIDOS des VEB Kombinat Robotron zum Einsatz (Gräßler 1987).

An diesem zweiten Beispiel lässt sich nachvollziehen, wie Änderungswünsche konkret in Beschlussvorlagen aufgenommen wurden. Es zeigt einen Partei- und Staatsapparat, der um Lösungen und internen Interessenausgleich bemüht war, der Sachverstand für seine Entscheidungsprozesse akquiriert, jedoch auch dadurch gekennzeichnet ist, dass übergeordnete Funktionäre die endgültigen Entscheidungen trafen und Beschlüsse fassten.

Der Historiker Rüdiger Bergien stellt in seiner Dissertation zur Organisationskultur und Herrschaftspraxis im Zentralkomitee der SED die These auf, »dass sich die Parteiherrschaft nicht als statischer Zustand fassen lässt [...]. Parteiherrschaft wird hier vielmehr als ein dynamisches Beziehungsgefüge zwischen Individuen, Gruppen und Institutionen verstanden« (Bergien 2017, 17). Er plädiert dafür, die SED nicht so sehr als »Transmissionsriemen« der den Willen der Partei »nach unten« weiterleitet« (ebd. 14) zu befragen, sondern sie als Teil der DDR-Gesellschaft zu begreifen, als nicht allein übergeordnet. Der an der TU Dresden tätige Technikhistoriker Thomas Hänseroth wies in einem Aufsatz 2010 darauf hin, dass sich in der DDR Wissenschaft und Staat gegenseitig befruchtet hätten, selbst wenn keine 100%ige Übereinstimmung in ideologischen Fragen bestand. Damit erteilt er vereinfachenden Positionen, demnach Veränderungen allein vertikal von oben nach unten gesteuert worden wären, eine Absage, denn die Parteifunktionäre seien keineswegs so »omnipotent« gewesen, wie sie es selbst gern gesehen hätten. »Tatsächlich aber waren weder sie noch die politische Elite in demokratisch verfassten Staaten in der Lage, fachwissenschaftliche Schwerpunkte vorzugeben. Vielmehr erfolgte dies zunächst über Aushandlungsprozesse auf der Ebene der wissenschaftlichen Akteure. Über diverse Expertengremien traten diese dann den Weg »nach oben« an, um schließlich als staatliche Vorgabe wieder den Weg in die Niederungen des Wissenschaftsbetriebs zu finden« (Hänseroth 2010, 412f.). Die Argumentation Hänseroths für die Wissenschaften, lässt sich auch auf den Bereich der Forschung, Entwicklung und Produktion der Rechentechnik und der Datenverarbeitung in der DDR anwenden.

Damit sind die groben Fluchtlinien der institutionellen Planung und Lenkung auf Staats- und Parteebene in der DDR gezeichnet.<sup>210</sup> Der nächste Abschnitt begibt sich auf Betriebsebene, diskutiert jedoch

---

<sup>210</sup> Auf Betriebsebene ist die Verortung der Ingenieure im Verhältnis zum SED-Staat anhand dreier Studien genauer zu verfolgen: Der Historiker Rainer Lepsius fasst basierend auf Zeitzeugeninterviews die Handlungsspielräume der Generaldirektoren der Betriebe zusammen (Lepsius 1995, 357–359). Zur Interrelation von SED-Organisation und Betriebsleitung mit den Angestellten, die vorwiegend geistige Tätigkeiten im Betrieb ausführten, äußert sich (Hürtgen 2009, 96–124, 257f.). Für die Softwareentwicklung führte die Historikerin Dolores Augustine 1999 eine Befragung von 20 Beteiligten in der DDR durch und zeichnete sehr unterschiedliche Lebens- und Karrierewege nach. Obwohl im Prinzip gleichgestellt, erfuhren die Softwareingenieurinnen berufliche Benachteiligungen in den männlich dominierten Ingenieurskulturen. Es bestanden Entscheidungs-

nicht Fragen der Machtdiskurse, sondern soll einen Überblick über die komplexe Konstituierung des aus vielen Einzelbetrieben zusammengesetzten VEB Kombinat Robotron geben.<sup>211</sup>

### 3.1.2 Das Kombinat Robotron

Eine quellenkritische Gesamtdarstellung des Kombinat Robotron steht derzeit aus. Abweichend von der sonstigen Vorgehensweise dieser Arbeit, Tiefenschichten freizulegen, soll hier nur ein äußerst kurzer Überblick erfolgen. Ziel ist eine grobe historische Orientierung, die sich weitergehender Untersuchungen enthält, wobei vor allem gezeigt wird, aus welchen einzelnen Elementen das Kombinat Robotron zusammengefügt wurde, um einen einheitlichen Hersteller für Computer und deren Peripherie, sowie der zugehörigen System-Software in der DDR zu schaffen.

Die Geschichtsschreibung für diesen Themenkomplex erfolgt bisher hauptsächlich durch die involvierten Beteiligten. Zur Geschichte des Kombinats Robotrons berichtet der Gründungsdirektor des Großforschungszentrums Robotron und Mitglied des Forschungsrates des Ministerrates der DDR Gerhard Merkel in (Merkel 2005) und zur Forschung und Entwicklung im VEB Robotron in (Merkel/Junge/et. al. 2006). Zum Softwarehaus VEB Robotron Projekt Dresden berichtete dessen ehemaliger Direktor Hans-Jürgen Lodahl (Lodahl 2005). Weitere Erinnerungsberichte ehemaliger Robotron-Mitarbeiter sind gesammelt auf einer umfangreichen Website die durch die AG Rechentechnik des Fördervereins für die Technischen Sammlungen der Stadt Dresden unter dem Patronat des Stadtarchives Dresden zusammengestellt wurde (AG Rechentechnik 2020). Ausführliche Informationsquellen zu Robotron sind die Websites robotrontechnik.de, welche durch ein Autorenkollektiv ehemaliger Mitarbeiter herausgegeben wird (Autorenkollektiv 2020), und eser-ddr.de, die durch den ehemaligen Chefkonstrukteur des ESER Hanns-Georg Jungnickel verantwortet wird (Jungnickel 2013).

Eine allgemeine Darstellung über die Entwicklung der Mikroelektronik in der DDR liegt in der Studie über den Einfluss des Ministeriums für Staatssicherheit auf die Entwicklung der Mikroelektronik in der DDR durch Barkleit vor, herausgegeben durch das Hannah-Arendt Institut für Totalitarismusforschung Dresden (Barkleit 2000). Im gleichen Kontext erschienen Studien von Erich Sobeslavsky und Nikolaus Joachim Lehmann zur Frühgeschichte der elektronischen Datenverarbeitung in der DDR (Sobeslavsky 1996). Der stellvertretende Minister für Elektrotechnik und Elektronik und Regierungsbeauftragte für die Mikroelektronik Karl Nendel publizierte eine Autobiographie (Nendel/Pasch 2018). Auch in den

---

freiräume, ob man sich am politischen Programm des SED-Staates beteiligen wollte, oder sich obgleich beruflich engagiert in's Private zurück ziehen wollte (Augustine 1999).

<sup>211</sup> Eine ausführlichere Darstellung würde sich neben Bergien auf eine ganze Reihe weiterer für die vorliegende Arbeit relevanter Studien und Untersuchungen stützen. Die Historikerin Monika Kaiser untersucht in *Machtwechsel von Ulbricht zu Honecker. Funktionsmechanismen der SED-Diktatur in Konfliktsituationen 1962 bis 1972* den Zeitraum und die Umbrüche, welche sich mit der in Fußnote 206 nach Gerhard Merkel zitierten Periodisierung überschneiden. In Abschnitt 2 verfolgt sie *Das Wagnis der Wirtschaftsreform 1962 bis 1966* und beschreibt, dass die Idee einer zunehmenden Eigenwirtschaftlichkeit der Betriebe mit den Eigeninteressen der Wirtschaftsleitung, namentlich Günther Mittags, über wirtschaftlichen Zugriff auch Macht ausüben zu können, kollidierte (Kaiser 1997, 57–85). Einen wichtigen Einblick in die Einbettung des Rechentechnik-Programms in das reformerische Neue Ökonomische System (NÖS) Mitte der 1960er Jahre gibt (Sobeslavsky 1996, 62–68). Der Sozial- und Wirtschaftshistoriker André Steiner stellt die Voraussetzungen und verschiedenen Phasen der NÖS historisch dar und konstatiert: »Alles in allem blieben die Marktelemente auch in der Wirtschaftsreform der DDR immer mit der Planung verbunden und wurden schließlich von ihr dominiert. Sie sollten die Planung ökonomisieren« (Steiner/André 2004, 62). Die *Diskussion der ökonomischen Krisen in der Führungsspitze der SED* in der Ära Erich Honecker stellt der Historiker Hans-Hermann Hertle in den Mittelpunkt seiner Überlegungen, und kennzeichnet das Mikroelektronik-Programm der 1980er Jahre als ein Ringen um Sieg oder Niederlage (Hertle 1995, besonders: S. 332–336).



Tagungsbänden *Informatik in der DDR – eine Bilanz*, herausgegeben durch Gabriele Schade, Professur für Medieninformatik an der TH Erfurt und den Historiker Friedrich Naumann, an der TU Chemnitz tätig, (Naumann/Schade (Hrsg.) 2006), sowie *Informatik in der DDR - Grundlagen und Anwendungen*, herausgegeben durch die an der TU Dresden tätige Informatikerin Birgit Demuth (Demuth/Gesellschaft für Informatik (Hrsg.) 2008) sind vorwiegend Zeitzeugen als Autoren vertreten.

Abweichend davon sind die quellenkritischen Beiträge von (Donig 2006; Donig 2009), und eine biographisch orientierte Studie zur Person Werner Hartmanns, als Generationsvertreter der frühen DDR-Mikroelektronik durch Dolores L Augustine (Augustine 2003), sowie darauf aufbauend die Monographie *Red Prometheus – Engineering and Dictatorship in East Germany, 1945–1990* (Augustine 2007) zu nennen, sowie im weiteren Kontext Ben Peters' *How Not to Network aNimation – The Uneasy History of the Soviet Internet* (Peters 2017).

Als Fokuspunkt dient die Gründung des VEB Kombinat Robotron am 1. April 1969. Das Kombinat kombinierte Volkseigene Betriebe (VEB) zu einem integrierten Wirtschaftsverbund, wobei die Zusammenarbeit über Binnenverträge geregelt war. Grob untergliedert war das Kombinat Robotron in drei Bereiche: Forschung, Produktion und Vertrieb.

1.) Den Bereich Produktion begründeten 1969 der Stammbetrieb VEB RAFENA Radeberg zur Produktion von Großrechnern, Kleinrechnern, Richtfunktechnik und Konsumgütern, die Großrechnerherstellung und später auch Produktion von Peripherie durch den VEB Elektronische Rechenmaschinen Karl-Marx-Stadt (ELREMA), das Ingenieurbüro für Rationalisierung Dresden und Betriebsteile zur Einzelteilproduktion, die in Riesa und Hoyerswerda ansässig waren. Zahlreiche Betriebe dienten dem Kombinat als Zulieferer, so das VEB Halbleiterwerk Frankfurt/Oder und der VEB Kontaktbauelemente Gornsdorf für elektronische Bauelemente, der VEB Carl-Zeiss Jena für Magnetbandgeräte und Plotter, der VEB Keramische Werke Hermsdorf als Speicherproduzent, der VEB Büromaschinenwerke Sömmerda für Lochkartentechnik und Drucker, und einige Betriebe der Nachrichtentechnik für Fernschreiber und Datenübertragungsmodule.

2.) Der Vertrieb verfügte über die Kapazitäten des VEB Bürotechnik Berlin für Beratung und Servicearbeiten und des VEB Robotron-Projekt Leipzig als Generallieferant für komplette Rechenzentren. Er war mit verschiedenen Schwerpunkten über Dresden, Leipzig, Berlin, Stralsund, Erfurt, Magdeburg, Halle/Saale und Karl-Marx-Stadt (heute Chemnitz) verteilt.

3.) Für Forschung und Entwicklung flossen in die Gründungskonstellation 1969 die Kapazitäten folgender Betriebe und Institutionen ein: Der VEB Elektronische Rechenmaschinen Karl-Marx-Stadt (ELREMA) erbrachte Forschungs- und Entwicklungskapazitäten für Großrechner und Betriebssysteme. Das Institut für Datenverarbeitung Dresden (idv) zeichnete verantwortlich für Industriecomputer für Fertigungsprozesse, sowie die dazugehörige Software und die Programmierung von Anwendungssoftware für Großrechner, sowie Informations- und Lehrmaterialien. Das Institut für Elektronik Dresden (IED) wurde als Entwickler und Produzent für Kernspeicher und Festplattenspeicher integriert. Das Institut für Maschinelle Rechentechnik (IMR) betrieb Grundlagenforschung im Bereich der Künstlichen Intelligenz, Schrifterkennung und Programmiersprachen. Die Forschungs- und Entwicklungsabteilung des VEB RAFENA Radeberg brachten Wissen über die Produktion von Großrechnern (dem Robotron R 300), Prozessrechnern und Kleinrechnern ein.

Aufgrund von Umstrukturierungen änderte sich mehrfach die Benennung der Forschungs- und Entwicklungsabteilungen. In Folge des Beschlusses des Politbüros des ZK der SED und des Ministerrates

der DDR von 1964<sup>212</sup> wurde nach Vorarbeiten und Vorstrukturierungen 1969 das VEB Kombinat Großforschungszentrum gegründet. Es beinhaltete verschiedene Forschungsinstitute und Produktionskapazitäten und war als Querschnittsforschungsbereich auch in Zusammenarbeit mit der Akademie der Wissenschaft und den Universitäten der DDR angelegt. Nach dem VIII. Parteitag der SED 1971, mit der Ablösung Walter Ulbrichts durch Erich Honecker, kam es zu einer Zäsur, durch welche das bis dahin betriebene ›Neue Ökonomische System der Planung und Leitung‹ (NÖS) nunmehr der ›Einheit von Wirtschafts- und Sozialpolitik‹ wich. Die Gremien der SED und die nachgeordneten Ministerien kürzten die Investitionsmittel für Rechentechnik spürbar und stuften das Großforschungszentrum zurück zum VEB Robotron Zentrum für Forschung und Technik (ZFT, 1971 bis 1984).

Anfang der 1980er Jahre spiegelten sich erneute Modernisierungsbemühungen in erneuten Strukturveränderungen des Kombinates wieder. Diese trafen auf eine Korrektur der Forschungskürzungen von 1971, welche in die CAD/CAM-Beschlüsse 1984 mündeten und die Digitalisierung der DDR durch Mikroelektronik und Personalcomputer vorantreiben sollten.<sup>213</sup> Die Exportfähigkeit Robotrons sollte gestärkt werden, und die zunehmende Bedeutung der Softwareherstellung und -pflege formalisiert werden. Während 1984 ein Großteil des Robotron Zentrum für Forschung und Technik (ZFT) in den forschenden und produzierenden VEB Robotron-Elektronik Dresden (RED) übergang, wurde die anwendungsorientierte Softwareentwicklung in den VEB Robotron-Projekt Dresden (RPD) ausgegliedert.<sup>214</sup> Die Entwicklung von Datenbanksystemen ist daher mit den beiden Namen VEB Robotron Zentrum für Forschung und Technik (ZFT) und ab 1984 mit VEB Robotron-Projekt Dresden (RPD) verknüpft. (Merkel 2005; Merkel/Junge/et. al. 2006).

Durch die zahlreichen Untergliederungen und Restrukturierungen des Kombinates entstand ein komplexes betriebliches Gebilde, welches sowohl horizontal als auch vertikal integriert war, aber keineswegs so monolithisch funktionierte, wie es der Name ›Kombinat Robotron‹ auf den ersten Blick aufruft, und es ursprünglich durch die Planungsgremien intendiert war. Umstellungen und Änderungen zeigen Ansätze, auf Krisen und Probleme der ökonomischen Situation in der DDR zu reagieren. Die Komplexität des Vorhabens wird brisanter vor dem Hintergrund der im vorherigen Abschnitt formulierten ›Kleines-Land-These‹, und lässt retrospektiv deutlich werden, dass die in der DDR verwendeten Planungsinstrumentarien mit Emergenz und Kontingenz nur unzureichend umgehen konnten.

---

<sup>212</sup> Vgl. *Programm von Maßnahmen zur Entwicklung, Einführung und Durchsetzung der maschinellen Datenverarbeitung in der DDR in den Jahren 1964 bis 1970* (Ministerrat der DDR/Politbüro des ZK der SED 1964).

<sup>213</sup> Vgl. *Grundlinie zur weiteren Beschleunigung der Entwicklung, Produktion und Anwendung der Mikroelektronik in der Volkswirtschaft der DDR* (Ministerrat der DDR/Politbüro des ZK der SED 1982), *Bericht über die Ergebnisse und die weiteren Maßnahmen bei der Gestaltung eines volkswirtschaftlich effektiven Produktions- und Exportprofils auf dem Gebiet der Rechen- und Bürotechnik im VEB Kombinat Robotron* (Ministerrat der DDR 1983), *Maßnahmen zur Beschleunigung der Entwicklungsarbeiten für die erforderliche Gerätetechnik und von Systemunterlagen sowie zur Erhöhung der Produktion von Geräten zur automatisierten Produktionsvorbereitung* (Ministerrat der DDR 1984).

<sup>214</sup> Teil dieser Umstrukturierung war eine Ausdifferenzierung der Computerproduktion in der DDR im Zuge der Mikroelektronik seit Mitte der 1970er Jahre. So wurde 1978 das Kombinat Mikroelektronik Erfurt (KME) gegründet, zu dem auch das in Dresden ansässige Zentrum für Forschung und Technologie Mikroelektronik Dresden (ZFTM) als Chiphersteller und der VEB Halbleiterwerk Frankfurt/Oder gehörte. Das Kombinat Carl-Zeiss Jena spielte ab den 1980er Jahren eine zunehmende Rolle als Spezialgeräte- und Maschinenausstatter für die Halbleiterherstellung und ab Mitte der 1980er auch als Hersteller von Mikroschaltkreisen, vgl. (Barkleit 2000, 20–33; Augustine 2003).

### 3.1.3 Embargo des Coordinating Committee on Multilateral Export Controls

Zu den oft erwähnten Rahmenbedingungen der Entwicklung von Hochtechnologie in der DDR zählte das Technologieembargo des Coordinating Committee on Multilateral Export Controls. COCOM war ein informeller Zusammenschluss eines Großteils der NATO-Mitgliedsstaaten plus Japan, initiiert 1949 durch die USA, in welchem Exportbeschränkungen für die (gegnerischen) Länder des Rats für gegenseitige Wirtschaftshilfe (RGW) beschlossen wurden. In geheimen Beratungen der Mitglieder entschieden sie Listen mit Embargogütern. Die beteiligten Länder setzten sie in nationales Recht um, die USA zum Beispiel in den Export Control Act von 1949 und Großbritannien in den UK Export Controls. Großbritannien zum Beispiel unterteilte die Embargo-Güter in drei Bereiche:

- Waffen, Munition, und allgemeine Militärgüter,
- Materialien und Ausrüstung für Nuklearforschung, Kernkraft und Kernwaffen,
- Industriegüter, und darin enthalten: Computer, Software, Telekommunikation, und Fototechnik.

Für geplante Exporte in die Länder des RGW waren entweder nationale Exportlizenzen oder COCOM-Exportlizenzen über eine gemeinsame Schiedsstelle in Paris zu beantragen, ein Prozess der von einem bis mehreren Monaten in Anspruch nehmen konnte (Van Kerckhove 1991, 9f.).

Das COCOM-Embargo sollte als politisches Instrument im Kalten Krieg die Lieferung von Hochtechnologien und Militärmaterialien an die sozialistischen Länder unterbinden. Die Embargo-Politik unterlag dementsprechend Schwankungen, welche sich aus den Eigeninteressen innerhalb der Embargoländer und der übergeordneten Lage zwischen den politischen Blöcken ergaben, zum Beispiel durch den Einstieg der UdSSR in den Afghanistan-Konflikt zu Beginn der 1980er, der eine erneute Verschärfung zuvor gelockerter Regulierungen nach sich zog (vgl. Hill 1991, 161f.; Donig 2006, 255).

Im Juli 1984 verabschiedete COCOM angesichts des Entstehens einer neuen, auf Mikrochips basierenden Computergeneration nach zweijährigen Verhandlungen aktualisierte Richtlinien für Personalcomputer und Software.<sup>215</sup> Jegliche Software, die zu einem unter Embargo stehenden Gerät gehörte, war ebenfalls vom Embargo getroffen. Das bedeutet, dass das Betriebssystem OS/360 ebenso wie die Hardware des IBM 360 nicht geliefert werden durfte. »Software kann auch eigenständig dem Embargo unterworfen werden [...]. Beispiele sind bestimmte Arten von Computer-Aided-Design (CAD)-Software, Software bezüglich der Sicherheit von staatlich geheim gehaltenen Materialien, bestimmte Arten von Hochtechnologie in Entwicklung, Programmierung, Diagnostik oder Maintenance, und Betriebssystemen. [...] Software die Public Domain [F.H.: gemeinfrei] ist, oder kommerziell verfügbare Standardsoftware unterliegen nicht dem Embargo« (Van Kerckhove 1991, 11).<sup>cxliii</sup> Nach diesen Regelungen dürften Datenbanksysteme grundlegend zu den Produkten gehört haben, welche dem Embargo unterlagen. Das Beispiel des schwedischen MIMER-Datenbankmanagementsystems zeigt jedoch die Löchrigkeit des Embargos (siehe Abschnitt 3.1.5 *Übersicht über Datenbanksysteme in der DDR*).

Die Auswirkungen für die DDR und die anderen am Einheitlichen System Elektronischer Rechner (ESER) beteiligten Länder waren spürbar, denn benötigte Hardware und Software konnten nicht aus dem Westblock importiert werden. Problematischer war, dass keine legale Lizenzproduktion erfolgte,

---

<sup>215</sup> Angesichts der Komplexität der Güterproduktion, sich verändernder Regelungen, teilweiser Lockerungen, der Eigenentwicklungen im Rahmen des ESER, und der Bemühungen zur Umgehung des Embargos durch Westlieferanten und Ostimporteure ist es kein Wunder, dass immer wieder Embargogüter in die RGW-Länder gelangten. Das Stellwerk der Reichsbahn Halle (RDZ) verfügte beispielsweise ab den 1970er Jahren über einen westlichen Kleincomputer PDP-11/44, der durch die Staatssicherheit über Österreich importiert worden war und welcher bis Januar 2015 dort im Einsatz war (Robotrontechnik.de 2019b).

da die Lizenzen für bestimmte Hardware einfach nicht erworben werden konnte.<sup>216</sup> So beschafften der Bereich Kommerzielle Koordinierung im Ministerium für DDR-Außenhandel, geleitet durch den Offizier im besonderen Einsatz (Oibe) des Ministeriums für Staatssicherheit (MfS) Alexander Schalck-Golodkowski, und die Abteilung Hauptverwaltung Aufklärung (HVA) V2 des MfS,<sup>217</sup> und der befreundete sowjetische Geheimdienst KGB teils legal, teils illegal unter Umgehung des Embargos über Drittländer, und teils durch klassische Wirtschaftsspionage IBM-kompatible Hardware, Software und Dokumentation als Vorlagen zum Nachbau (Vogel u. a. 1994; Barkleit 2000, 19; Der Bundesbeauftragte für die Unterlagen der Stasi der DDR (Hrsg.) 2013, 111, 231, 234).

Das COCOM-Embargo war allerdings nicht alleiniger Grund. Potentiellen Kooperationen widersprachen oft auch die DDR-eigenen übergeordneten Organe. Gerhard Merkel, Gründungsdirektor des VEB Robotron und zeitweise Mitglied im Ministerrat der DDR, berichtete beispielsweise, dass nach langen Verhandlungen mit Konrad Zuse Mitte der 1960er Jahre eine geplante Kooperation mit dessen Firma nicht zu Stande kam, da einerseits die Regierung in Bonn aufgrund des Embargos ablehnte und andererseits die SED-Parteizentrale eine Zusammenarbeit aus innenpolitischen Erwägungen absagte (Interview Merkel 2018, Fol 2f.).

Es wäre falsch, das COCOM-Embargo als einzigen Grund für die Probleme der Technologieentwicklung zu lesen. Es war ein Faktor, aber nur einer von mehreren. Der Historiker Simon Donig argumentierte 2006, »daß die große Bedeutung, die dem Embargo zugeschrieben wird, nur in bestimmten Grenzen gelten kann und daß generell die Bedeutung exogener Faktoren für die Schwierigkeiten der DDR bei der Technogenese nicht überbewertet werden darf« (Donig 2006, 251f.). Das COCOM-Embargo wird daher hier nicht als Begründung, sondern als Kontext herangezogen. Illustrieren lässt sich dies an dem Umstand, dass die DDR zwar über eine Mikrochip-Produktion verfügte, diese aber nicht rentabel war, da die Stückzahlen viel zu gering blieben. Die Möglichkeiten des Nacherfindens von Technologie, wie sie für die auf Transistortechnologie basierten IBM-Computer im Rahmen der ersten und zweiten ESER-Generation noch sinnvoll war, erwies sich mit dem Umstieg auf Mikrochips als technologisch und organisatorisch nicht mehr beherrschbar. Weil der potentielle maßgebliche Partner, die UdSSR, nur unzureichende Quantitäten und Qualitäten lieferte, wurde unter Aufwendung massiver Investitionsmittel in der DDR versucht, eine eigene Microchip-Herstellung aufzubauen. Dies gelang in Ansätzen, generierte eine große Menge eigener Expertise, allerdings erwies sich die Herstellungs-Technologie letztlich als hochkomplex und äußerst ressourcenintensiv (vgl. Barkleit 2000, 26, 83–92) für ein Land von der Größe der DDR. Der Einzelpreis der schließlich produzierten Chips war im Weltmarktvergleich zu hoch.<sup>218</sup> Gleichzeitig blieb es der DDR durch das Embargo, aber auch die eigenen begrenzten Valuta-Mittel für Importe verwehrt, Mikrochips für Computer, Taschenrechner oder digital gesteuerte Werkzeugmaschinen käuflich zu erwerben. Eine Lizenzproduktion amerikanischer Technologie wie durch Siemens in der BRD war nicht möglich. Insofern trug das Embargo zur als problematisch empfundenen Lage der Hochtechnologie-Innovation ebenso bei, wie der Unwille des Partners UdSSR,

---

<sup>216</sup> Protagonisten von Hochtechnologie führen das COCOM-Embargo als Begründung an, warum die Produkte nicht lizenziert werden konnten. Dieser entlastende Begründungszusammenhang verdrängt, dass die verfügbaren Valutamittel der DDR zum Kauf von Lizenzen äußerst begrenzt waren, und somit Lizenzierung im Unterschied zur nahezu kostenneutralen Nutzung von Kopien oder zur Nachentwicklung häufig nicht in Frage kam.

<sup>217</sup> Lies: Römisch 5 Unterabteilung 2, bis 1962 Abteilung »Wissenschaftlich-Technische Auswertung«.

<sup>218</sup> Der Einzelpreis des 64k-DRAM-Chips, hergestellt durch das Kombinat Mikroelektronik Erfurt (KME), lag 1989 bei 40,00 DDR-Mark (Barkleit 2000, N. 64). Im Zuge eines Imports aus der BRD erwarb die DDR 1985 im Vergleich dazu 15.000 Speicherchips dieses Typs zum Stückpreis von 2,45 DM (ebd., N. 99).

die DDR an Schlüsseltechnologien zeitnah teil haben zu lassen. Zusammenfassend führte die Situation in der DDR zur drei Strategien:

- Erstens, wurde ein Teil der benötigten Waren unter Umgehung des Embargos durch Tarnfirmen des MfS importiert.
- Zweitens, forcierte man die Eigenentwicklung von Hochtechnologien im Hardware und Softwaresektor, darunter auch Datenbanken, was einerseits umfangreiche Ressourcen band, andererseits für eine hohen Forschungsstand sorgte.
- Drittens, erfolgte, wenn möglich, die Übernahme existierender Technologien und deren Anpassung auf die eigenen Rahmenbedingungen unter dem Stichwort der ›Adaption‹.

### 3.1.4 Software Adaptionen

Für die vorliegende Arbeit ist die Rahmenbedingung des COCOM-Embargos in zweierlei Hinsicht relevant:

- Erstens spielten für relationale Datenbanksysteme, in welchem die Zuweisung von Speicheradressen automatisiert werden sollte, die verfügbaren Computerressourcen eine wichtige Rolle. Automatisch angelegte Indexe realisieren die Automatisierung der Adressverwaltung, und diese bedurfte einer für den damaligen Stand höheren Rechenkapazität und Speicherbedarfs, als in netzwerkorientierten Systemen, in denen die Speicherverwaltung und -adressierung durch die Programmierer erfolgte. Da der DDR-Import aktueller Rechentechnik aus den Embargo-Ländern erschwert war, blieb die Entwicklung relationaler Systeme in der DDR an den nationalen Entwicklungsstand der Rechentechnik gebunden.
- Zweitens war es schwierig bis unmöglich, relationale Datenbanksysteme in Lizenz zu übernehmen. Daher verfolgte die DDR Strategien des Kopierens und Adaptierens von Software. Aus diesem Grund soll im Folgenden näher auf das Adaptieren als Praxis der Softwareprogrammierung eingegangen werden.

Einer der damals bei Robotron-Projekt involvierten Ingenieure, Jürgen Bittner, definiert Adaption wie folgt: »[Die] Adaption des originalen Programmcodes [erfolgte] durch Ersetzen der Namen aller Programmelemente wie Datenobjekte, Variablen, Konstanten, usw., und eigene Entwicklung einiger Zusatzfunktionen« (Bittner, in: Interview Bittner/Heinemann 2018, Fol 3). Es handelte sich bei Adaptionen jedoch nicht allein um eine reine Folge von Sachzwängen, vielmehr war und ist es eine übliche Form des Wissenstransfers, Produkte zu untersuchen und vielversprechende Konzepte zu übernehmen. »Die Entwicklung eines Sets von Spezifikationen für ein komplexes Hardwaresystem durch eine geordnete Untersuchung von Probeexemplaren dieses Systems« (Rekoff 1985, 244)<sup>cxliii</sup> ist spätestens seit den 1980er Jahren unter dem Begriff des ›Reverse Engineering‹ bekannt. Adaptieren ist daher eng verwandt mit anderen Praxen des Softwareschreibens. Beim ›Refactoring‹, erfolgt das Überarbeiten und Straffen vorhandenen Codes mit dem Ziel, durch Klarheit Fehlerquellen zu minimieren (Griswold 1991). Weitere Praxen sind das ›Forking‹, das Abwandeln vorhandenen Codes für einen anderen Entwicklungsstrang bei Verdoppelung der Code-Basis (Robles/González-Barahona 2012), und das ›Templating‹ beziehungsweise ›Cloning‹, das heißt der Code-Wiederverwendung im Rahmen von Code-Schnipseln oder Vorlagen (Rattan/Bhatia/Singh 2013, 1167). Beim ›Debugging‹ erfolgt die Fehlerkorrektur durch Speicherüberwachung und Parameterausgabe während eines Programmablaufes bzw. durch Auswerten von Fehlermeldungen im Anschluss an einen Programmablauf (Kidwell 1998). Bei der

›Disassemblierung‹ wird der ausführbare Maschinencode in für Menschen verständliche Sprachelemente, meist in Assembler, rückübersetzt (Schwarz/Debray/Andrews 2002). Diese kulturtechnischen Vollzüge der Programmierung sind seit den 1980er Jahren grundlegender Teil der Informatikausbildung und haben sich seit Mitte der 2000er Jahre mit der Zunahme kollaborativer Open-Source-Software verstärkt ausgebreitet.

Dass die Praktiken des Adaptierens keineswegs auf planwirtschaftlich agierende Länder begrenzt war, zeigt ein Blick in die Geschichte der Betriebssysteme für Personalcomputer. Die durch unmittelbare Übernahme des Codes unter Veränderung der Namensgebung erfolgende Adaption kann beispielhaft beobachtet werden für die Genealogie des Betriebssystems Microsoft DOS. Bill Gates versprach 1980 für IBMs ersten PC ein Betriebssystem zu liefern. Es erwarb aus Zeitgründen dafür *86-DOS/Quick and Dirty OS* (QDOS), welches der Programmierer Tim Paterson von dem Betriebssystem *Control Program for Microcomputers* (CP/M) adaptiert hatte. Das ursprünglich von Gary Kildall für die Firma Digital Research entwickelte CP/M lief nur auf einem Intel 8080-Prozessor und konnte daher nicht direkt auf dem IBM PC verwendet werden, der auf dem Nachfolger Intel 8086 basierte.

Tim Paterson übernahm die ersten 36 Funktionen von CP/M wobei er einige umbenannte, beispielsweise ›Read Sequential‹ in ›Sequential Read‹. Statt des ursprünglich von CP/M verwendeten Dateisystems nutzte er jedoch ein anderes, namens File Allocation Table (FAT). Nach eigener Aussage hatte Paterson nicht direkt Code kopiert, sondern die Funktionen, auch aufgrund des abweichenden Dateisystems, laut Bedienhandbuch nachempfunden. Ihm widersprechend schildert der Entwickler von CP/M Gary Kildall, wie auch eine Übernahme des Codes möglich gewesen sei: Ein Hilfsprogramm namens Dynamic Debugging Tool (DDT), ein sogenannter Disassembler erlaubte, den bereits kompilierten Code zu untersuchen, und CP/M teilweise oder ganz in die Ausgangssprache Assembler oder in Maschinencode zurück zu wandeln. Der rekonstruierte Code musste allerdings für die veränderte Hardware, z.B. einen anderen Prozessor, untersucht und angepasst werden (Evans/Buckland/Lefer 2004, 402–419).<sup>219</sup>

Beide Vorgehensweisen, Disassemblierung und Anpassung oder die Übernahme der Konzepte und deren Nachbildung fallen in der Praxis zusammen, die Bittner als Adaption bezeichnet. Microsoft Gründer Bill Gates verkaufte diese Adaption von CP/M aka QDOS an IBM, welche das nunmehr MS-DOS getaufte Betriebssystem mit den IBM PC auslieferte. CP/M war nicht nur Vorbild für MS-DOS, sondern auch für das DDR-Betriebssystem SCP, welches auf einer ganzen Reihe von Computern lauffähig und weitestgehend CP/M-kompatibel war. Dadurch konnte unter SCP auch westliche Software verwendet werden, die für gewöhnlich als nicht-lizenzierte Kopien kursierte (Robotrontechnik.de 2019a). In der DDR handelte es sich bei der Adaption von Software um eine weit verbreitete Aneignungsstrategie, die der Ressourcenknappheit des Landes entsprach. Einer der daran beteiligten Entwickler, Joachim Körner, bezeichnete den Prozess jedoch als »personalintensiv« (Körner 2012, Fol 4). Die CIA schätzte 1977 in einem Bericht über die Verwendbarkeit von IBM OS/360-Software in Bezug auf DDR-Großcomputer ein: »Nutzer des ostdeutschen [Robotron] EC-1040 [können] unmittelbar Anwendungsprogramme des IBM 360 verwenden. Ostdeutschland scheint das IBM-Betriebssystem erfolgreich dupliziert zu haben« (Central Intelligence Agency 1977, 7).<sup>cxliv</sup> Damit könne, so der Historiker Simon Donig die Strategie des ›Nacherfindens‹ in der DDR als »durchaus erfolgreich« gekennzeichnet werden (Donig 2006, 268).

---

<sup>219</sup> Dass dies kein Einzelfall war und ist, zeigt die Übernahme grundlegender Konzepte des Grafischen Benutzeroberfläche von XEROX Parc durch Apple Anfang der 1980er Jahre (Vgl. Ceruzzi 2003, 261–263).

Es stellt sich die Frage, warum die DDR überhaupt adaptierte oder ›nacherfand‹. Ein wichtiges Potential des ›Nacherfindens‹ war die Möglichkeit, Hardware und Software zu schaffen, welche in Drittländer exportiert werden konnte. Voraussetzung dafür war jedoch die »Schutzrechtsfreiheit für die auszuliefernde Software, deren Dokumentation und deren Verteilungsdatenträgern, entsprechend internationaler Rechtslage« (Körner 2012, Fol 3), was in der Praxis bedeutete, dass der komplette Code lizenziert oder selbst geschrieben sein musste. Nur Eigenentwicklungen konnten für dringend benötigte Valuta in das nicht-sozialistische westliche Ausland verkauft werden.

Andere Nacherfindungen, beispielsweise das Betriebssystem MOOS 1600 für den Kleincomputer K 1600, der auf einer DDR-eigenen Chip-Entwicklung basierte, waren nur teilkompatibel, sodass dafür gesonderte Software entstehen musste und konnte. Dem wird im Kapitel 3.3. *Kalifornien, Technische Universität Dresden und Robotron Dresden: Das Problemorientierte Seminar »Datenbanken« und DABA 1600* der vorliegenden Arbeit nachgegangen.

### 3.1.5 Übersicht über Datenbanksysteme in der DDR

Im Zuge der Recherchen wurde eine Vielfalt an DDR-Datenbanksystemen sichtbar. Daraus wird eine Auswahl näher diskutiert, welche sich an der Materialität der jeweiligen Systeme orientiert. Mit der Entscheidung, sich im Folgenden auf BASTEI, SOPS und DABA 1600 zu konzentrieren, bleiben einige DBMS zwangsläufig Desiderat. Anliegen der vorliegenden Arbeit ist es, jene Figurationen, die für bestimmte techno-soziale Entwicklungen stehen, herauszugreifen und eingehend nach ihrer Genese, Operationalität und Nutzung zu befragen. Diese Vorgehensweise lässt der vorliegenden Vielfalt zwangsläufig Unrecht widerfahren und umgeht Fragen der quantitativen Verbreitung, aufgrund derer AIDOS und DBS/R hätten in den Vordergrund gestellt werden müssen. Um trotzdem einen ersten Überblick zu gewährleisten, sind in der folgenden Tabelle die wichtigsten in der DDR verfügbaren Datenbanksysteme, ihre Anbieter, Herkünfte, Erstveröffentlichung und das zugrundeliegende Datenmodell aufgeführt (Abb. 51).

Kursorisch sollen aus dieser Übersicht einige ausgewählte DBMS Darstellung erfahren, die als Kontext relevant sind, jedoch vorrangig aus Zeitgründen keine eigene ausführliche Darstellung in Kapiteln fanden.

AIDOS: So fand das dialogorientierte Dokumentensystem AIDOS ab 1974 in der DDR große Verbreitung. Es enthielt Fähigkeiten zur Volltextrecherche in unstrukturierten Dokumenten, zur Indexierung, die Möglichkeit eines Thesaurus, eine eigene Abfragesprache, und in späteren Versionen Mustererkennungsverfahren zum automatisierten Indexieren von Dokumenten. Innerhalb des Thesaurus unterschied sich der Zugang in Deskriptoren und Synonyme. Die Synonyme bezogen sich auf die Deskriptoren. Zwischen den Deskriptoren waren eine Reihe von Relationen darstellbar: »begriffliche Unterordnung¶ Teil-Ganzes-Beziehung¶ assoziative Beziehungen¶ fünf beliebig interpretierbare symmetrische Beziehungen¶ fünf beliebig interpretierbare asymmetrische Beziehungen« (Hartmann 1987). AIDOS bestand aus einer Reihe von Modulen: Datenbeschreibung, Wörterbücher/Klassifikationen, Informationsspeicher und Sucherlebnisspeicher. Für die interne Datenverwaltung wurde ein B-Tree-Index in Kombination mit Schlüsselpaaren verwendet, eine Art der Organisation, welche auch in relationalen DBMS üblich war. AIDOS wurde für Leitungsinformationssysteme und Bibliothekssysteme auf unterschiedlichen Rechnergenerationen der DDR eingesetzt.

<b>Produktname</b>	<b>Hersteller/Vertrieb</b>	<b>Herkunft</b>	<b>Erstveröffentlichung</b>	<b>Datenlogik</b>
BASTEI – Bankspeicherung technischer Informationen	VEB Robotron ZFT	IBM BOMP-Adaption	1972	netzwerk-orientiert
SAWI – Speichern und Automatisiertes Wiederauffinden	VEB Robotron ZFT	Eigenentwicklung	1973 (?)	Dateiorientiert
AIDOS– Automatisiertes Informations- und Dokumentationssystem	VEB Robotron ZFT	Eigenentwicklung	ab 1974	Informationsrecherchesystem
DBS/R – Datenbank-system/Robotron	VEB Robotron ZFT	Eigenentwicklung basierend auf BASTEI und SAWI	1973–79	netzwerk-orientiert (Adressverkettung)
ESY/R – Exekutivsystem/Robotron	VEB Leitzentrum für Anwendungsforschung	Eigenentwicklung (?)	1975	Transaktionsmanager für Mehrnutzerbetrieb an DBS/R
DAKS – Datenkommunikationssystem	VEB Robotron	IBM CICS-Adaption	Ende der 1970er	Transaktionsmanager für Mehrnutzerbetrieb an DBS/R
MIMER	Hochschule Uppsala Schweden	Lizenz-Import und Weiterentwicklung	1982	relational
DABA 1600 – Datenbank 1600	TU Dresden und VEB Robotron Projekt	Eigenentwicklung	1984	relational
Redabas (auch TQL-Base)	VEB Robotron Projekt, VEB EAW Berlin	dBASE II und dBASE III-Adaption	1985	relational
ALLDBS	VEB Robotron Projekt	Oracle 5.1-Adaption	1985	relational
TOPAS	VEB Leitzentrum für Anwendungsforschung	Eigenentwicklung	1985	relational
ING_DB (auch INDES, DABA 32)	Zentralinstitut für Kybernetik und Informationsprozesse der AdW	Ingres 6-Adaption	1986	relational
WEGA-DATA	EAW Berlin Teltow	Unify 2000-Adaption	1986	relational
INTERBAS	VEB Robotron Projekt und Zentrprogrammssystem, Kalinin	Eigenentwicklung, Joint Venture	seit 1987, 1990 abgebrochen,	relational, netzwerk-orientiert, objektorientiert

Abb. 51: In der DDR verfügbare Datenbanksysteme. Als Jahr der Inbetriebnahme wird die frühe Phase der Produktverfügbarkeit angegeben, das heißt das Publikationsjahr der Software-Dokumentation. Alle Angaben beziehen sich nur auf die erste Version, spätere Überarbeitungen oder Portierungen auf andere Rechnersysteme sind nicht gesondert vermerkt (Autor).

Im Jahr 1987 wurden Versionen für 8-bit Rechner und 16-Bit Mikrorechner veröffentlicht und ebneten den Weg für den individuellen Gebrauch auf Personalcomputern. Auf einer AIDOS-Konferenz die 1989 in Leipzig stattfand, gab es Teilnehmende aus 79 Institutionen und Betrieben, und die Anzahl der



AIDOS-Installationen wurde auf 150 geschätzt. Nach 1990 wurde AIDOS zuerst in die Firma Data-Print Robotron System GmbH ausgegründet, um weitere Komponenten erweitert, und 2007 AIDOS Aktiengesellschaft überführt, welche ein Jahr später insolvent ging (vgl. VEB Robotron 1974a; Stys 1981; Hartmann 1987; Lorenz 1989; Hartmann/Arendt/Hans-Dieter u. a. 2008).

DBS/R: Das Datenbanksystem/Robotron (DBS/R) war mit über 250 Installationen das in der DDR auf Großrechnern am meisten verwendete Datenbanksystem. Es wurde durch das Robotron ZFT unter Leitung von Rolf Heinemann und fachlicher Leitung von Jürgen Bittner basierend auf den Erfahrungen mit den Datenbanksystemen BASTEI und SAWI ab 1971 entwickelt und stand ab 1973 zur Verfügung. DBS/R, welches nach dem Netzwerkdatenmodell organisiert war, erreichte einen Großteil der Anforderungen des CODASYL-Modells, ohne direkt als CODASYL-Datenbank konzipiert worden zu sein.<sup>220</sup> Durch die Einbindung von Transaktionsmonitoren wie ESY/R und DAKS war ein Mehrnutzerbetrieb mit Terminalzugriff möglich, und ab Version 6, die 1987 erschien, stand ein integrierter Transaktionsmanager zur Verfügung, der erhebliche Performancevorteile ermöglichte. DBS/R war auf den Stapelbetrieb für DDR-Großrechner konzipiert und ermöglichte die zeilenweise Eingabe und Ausgabe, ein graphisches User-Interface war nicht vorgesehen. Anwendung fand DBS/R in der Industrie, hauptsächlich in Elektrotechnik/Elektronik, Erzbergbau, Maschinen- und Fahrzeugbau, in der Chemischen Industrie, aber auch in Verwaltung, Postwesen und Hochschulen. Exportlösungen existierten in der UdSSR, Ungarn, China und Indien. Gegen der Ende der 1980er Jahre, nach Veröffentlichung der Version 6, neigte sich die aktive Entwicklung von DBS/R durch Robotron Projekt Dresden dem Ende zu, da sich mit dem System INTERBAS bereits eine Nachfolgerin in Entwicklung befand (Autorenkollektiv/Gräßler/Freudenberg 1975; Kosche 1980; Bittner 1984; Bittner 1986a; Bittner 2008; Heinemann, in: Interview Bittner/Heinemann 2018, Fol 2). DBS/R war eines der Datenbanksysteme, welches neben programmeigenen Datenstrukturen und BASTEI in Verbindung mit der modularen Informationsmanagementsoftware SOPS verwendet wurden, auf die ausführlicher in Kapitel 3.2 *Robotron und SAP: eingegangen* wird.<sup>221</sup>

DAFEMA: Das Datenbankbetriebssystem mit Fernzugriff Magdeburg (DAFEMA) wurde seit 1978 im VEB Datenverarbeitungszentrum Magdeburg als Zentrales Auskunftssystem der DDR-Post entwickelt und auf einem ESER-Großrechner eingesetzt. »Das Programm erfordert eine Speichergröße von 2,5 KByte, und einen Speicher von 4 KByte« (Waldmann 1981, 103). Der Zugriff war über das

---

<sup>220</sup> Zu CODASYL ausführlich siehe (CODASYL Data Base Task Group 1969a; CODASYL Data Base Task Group 1969b; Olle 1981).

<sup>221</sup> Die Bedeutung von DBS/R kann in der vorliegenden Arbeit nur unzureichend dargestellt werden. Andeutungsweise sei auf die zahlreichen Veröffentlichungen in der Zeitschrift *rechentechnik/datenverarbeitung* verwiesen, die sich mit DBS/R auseinandersetzen. Die Autoren vom Institut für Energetik Leipzig, Helga Haubold und Reinhard Fiedler hoben 1976 beispielsweise die Bedeutung für die Kohle- und Energiewirtschaft hervor: »Ziel für die gesamte Volkswirtschaft ist es, ein besseres Verhältnis von Aufwand und Ergebnis zu erreichen, d.h. zu einer höheren Effektivität zu gelangen. Ein Instrument zur Erreichung dieses Zieles ist die permanente Analysentätigkeit. [...] Die Datenbank mit dem Datenbankbetriebssystem DBS/R bietet dafür die systemtechnische Voraussetzung« (Fiedler/Haubold 1976, 48). Rolf Heinemann, Leiter der Abteilung Datenbankentwicklung/Robotron ordnete DBS/R als netzwerkorientiertes DBMS nach dem CODASYL-Modell ein: »Vertreter dieses Modells sind solche Systeme wie TOTAL von CINCOM SYSTEMS, SETOR von INTERPROGRAMMA Sofia, UDS von Siemens, IDMS von ADV/ORGA und im bestimmten Umfang auch DBS/R von Robotron« (Heinemann 1985, 108). Zahlreiche weitere Beiträge in *rechentechnik/datenverarbeitung* und in *Neue Technik im Büro*, sowie einschlägige Buchveröffentlichungen zeigen die Bedeutung von DBS/R an.

Programmierbare Terminal (PBT 4000) im ›Dialogbetrieb‹ als Stapelverarbeitung unter Einsatz von Transaktionen möglich, und zwar per Standleitung oder auch Einwahlleitung.

Zu den Anwendern gehörten beispielsweise das VEB Petrochemische Kombinat (Schwedt), welches seinen Artikelkatalog in Magdeburg verwaltete, die Bezirksfernmeldeämter zur Telefonnummern- und Vorwahlauskunft und zur Datenverwaltung für die gedruckten Telefonbücher. Zu den weiteren Anwendungen zählten Artikellisten im Maschinenbau und der chemischen Industrie, sowie ein Warenwirtschaftssystem im DDR-Warenhandel. Für die 1989 verwendete Version ist von einer relationalen Datenorganisation die Rede, bei welcher eine einfache Datenmanipulationssprache in andere Host-Sprachen (z.B. Assembler, C, PL/1) eingebettet werden konnte. In den 11 Jahren seiner Verwendung wurde das System kontinuierlich weiter entwickelt, sodass es 1989 auch für den Zugriff über den Robotron PC 1715 zur Verfügung stand und über zahlreiche Annehmlichkeiten wie Dateneingabe per Formularmaske, eine Anknüpfung an Redabas, die Ausführung vorabgespeicherter Prozeduren oder automatisches Backup verfügte. Bis 1989 erfuhr DAFEMA circa 60 Installationen in der DDR (Riewendt 1980; Waldmann 1981; VEB Datenverarbeitungszentrum Magdeburg (Hrsg.) 1987; Ruhland/Voss/Schreiber 1989).<sup>222</sup>

MIMER: Das relationale MIMER wurde von der Universität Uppsala/UDAC seit den späten 1970er Jahren entwickelt. Im Zuge einer Kooperationsvereinbarung, die zwischen UDAC und der Medizinischen Akademie Dresden getroffen wurde, konnten sie eine Lizenzvereinbarung treffen und MIMER war als erstes relationales DBMS in der DDR ab 1982 verfügbar. Diese Kooperation beruhte auf Gegenseitigkeit, so lieferten die an der Medizinischen Akademie unter der Leitung von Uwe Wloka beschäftigten Entwickler beispielsweise die deutschsprachige Übersetzung, einen Report-Generator und ein Zahnärzte-Fachsystem nach Schweden. Ein Großteil der Tätigkeit der Dresdener Entwickler bestand in einer Anpassung des für IBM/360 und später PDP-11 entwickelten Codes, auf die Hardware der kompatiblen ESER, respektive SKR-Computer in der DDR. Eingesetzt wurde es in den Krankenhäusern der DDR für Patienteninformationssysteme, aber auch für ein zentrales Krebsregister der DDR, oder ein Arzneimittelinformationssystem, aber auch in Industriebetrieben, welche MIMER über die Medizinische Akademie Dresden in Valutamark erwerben konnten. Industrieanwender waren die Reichsbahn/DDR, Abteilung Güterverkehr, der VEB Flugzeugwerft Dresden oder Chemische Betriebe. Weitere Verwendung fand es im Forschungs- und Verwaltungsbereich an Universitäten. Insgesamt gab es 106 Nachnutzungen in der DDR. Mit den Strukturänderungen in Folge des Endes der DDR und der zunehmenden Dominanz anderer relationaler Datenbanksysteme endete 1992 die konkrete Arbeit an MIMER (Schreiter/Stange/Vater 1982; Schreiter u. a. 1987; Wloka 2019). Das Produkt MIMER SQL wird mit Stand Ende 2019 weiterhin durch die schwedische Mimer Information Systems entwickelt und vertrieben (Eck 2011).

Redabas (Relationales Datenbanksystem). Das semi-relationale Redabas zählte zu jenen Datenbankmanagementsystemen die auf Personalcomputern lauffähig waren, und Teil dessen wurden, was Kevin Driscoll als ›Datenbankpopulismus‹ in den 1980er und 90er Jahren bezeichnete (Driscoll 2012, 18–26). Redabas verbreitete sich ab 1985 in der DDR mit der Einführung der Personalcomputer, zum Beispiel der Robotron A51xx-Serie und des PC 1715, da es das Erstellen eigener datenbasierter Softwarelösungen jenseits der Großrechenzentren ermöglichte. Redabas basierte maßgeblich auf dem westlichen

---

<sup>222</sup> In der Literatur wird DAFEMA als Eigenentwicklung dargestellt, es könnte sich jedoch auch um eine Adaption handeln. Darauf weisen u.a. englischsprachige Anweisungen der Datenmanipulationssprache wie „HLP“ (Help), „SAVE“ hin.

Produkt Dbase II (und später Dbase III) der Firma Ashton-Tate. An seiner Adaption war ein kleines Teil-Team, bestehend aus Ursula Hempel, Hans Lolei, Rita Winkler und weiteren für zwei Jahre beteiligt, da wesentliche Ressourcen der Softwareschmiede Robotron Projekt in das Datenbankprojekt INTERBAS

flossen (Heinmann in: Heinemann/Bittner, Fol. 36). Es gelang, aus dem Westen den Quellcode von Dbase II illegal zu besorgen, der für die Robotron-Computer angepasst und mit neuem Namen versehen wurde. Das Programm wurde durch Robotron Projekt Dresden bis 1990 in 15.000 Anwendungsfällen verkauft (ebd.). In der DDR war es ein offenes Geheimnis, dass Redabas im Großen und Ganzen dem westlichen Produkt dBase entsprach. Nach 1990 ermöglichte dies einen fast nahtlosen Umstieg auf das originale dBase und es gibt Datenbankanwendungen, welche bis Ende der 1990er Jahre auf einem DOS-Emulator unter Windows betrieben und weiterentwickelt wurden (Uhlig 2019, Fol 10f.). Redabas war tabellenorientiert und wurde daher als ›relational‹ vertrieben, allerdings verfügte es nur über ein gewisse Schnittmenge mit wirklich relationalen DBMS. Dies tat seiner Popularität keinen Abbruch. Buchpublikationen (Hempel/Loley 1985; Hempel/Loley 1988; Hempel/Loley 1989), zahlreiche Beiträge in Rechentechnik/datenverarbeitung zeugen davon ebenso (Hempel/Loley 1985; Kruczinsky 1987; z.B. Redaktion 1989) wie Fernsehsendungen des DDR-Bildungsfernsehens, welche Redabas vorstellten (Lehmann 1987).

Dieser kurze Überblick hat vor allem jene eigenentwickelten DBMS der DDR betrachtet, welche nicht ausführlicher in den folgenden Kapiteln Beachtung finden konnten. Zur weiteren Orientierung sollen hier jene DBMS erwähnt werden, welche eingehend diskutiert werden: Das frühe DBMS BASTEI wird ausführlicher in Abschnitt 3.2.1.1 *Explosionen: Bankspeicherung Technischer Informationen (BASTEI) / Bill Of Materials Processor (BOMP)* diskutiert. Der relationalen DDR-Eigenentwicklung DABA 1600 widmet sich das Kapitel 3.3 *Kalifornien, Technische Universität Dresden und Robotron Dresden: Das Problemorientierte Seminar »Datenbanken« und DABA 1600*, in welchem das durch das Ende der DDR abgebrochene Großprojekt INTERBAS erwähnt wird.

## 3.2 Robotron und SAP: Sozialistisches und kapitalistisches Enterprise Resource Management im Vergleich

»Der Aufstieg der relationalen Datenbanken war ein wichtiges Phänomen der Unternehmens-Software in den 1980er und 1990er Jahren. [...] Eine dritte Kraft, welche die Softwareindustrie für Unternehmen formte, war Enterprise Resource Planning Software, das Phänomen der Anwendungssoftware der 1990er« (Campbell-Kelly 2003, 169).<sup>cxlv</sup>

Die modular aufgebauten Sachgebietsorientierten Programmiersysteme (SOPS), so rückschauend Rolf Gräßler, seien das »SAP des Ostens« gewesen. Lohnt es sich, diesem interpretativen Rahmen der Systemkonkurrenz zwischen Kapitalismus und Sozialismus zu folgen und SOPS und SAP einander gegenüber zu stellen? Was folgen würde, wäre eine Technikgeschichtsschreibung des ›Wer war Erster?‹. In der vorliegenden Studie soll der Vergleich von SOPS und SAP anders gelesen werden: Unter anderen politischen und ökonomischen Vorzeichen als im Westblock waren SOPS eines der frühen integrierten Systeme aus denen die späteren kulturellen Techniken und Technologien des Enterprise Resource

Planning (ERP), die Planung wirtschaftlicher Ressourcen in einer Organisationseinheit, des Customer Relationship Management (CRM), das heißt Kundenbetreuung, Marketing und Absatzplanung, und des Supply Chain Management (SCM), der Material- und Warenlogistik, hervorgegangen sind. Im Westblock der 1960er Jahre wurden diese Vorgängersysteme unter der Bezeichnung Material Requirements Planning (MRP) bekannt, und aus dieser Perspektive sollen auch SOPS diskutiert werden.<sup>223</sup> Die Kernfunktionalität von MRP-Systemen konzentrierte sich auf die Lösung der Stücklistenfunktionalität. Diese stand historisch an der Schwelle einer Entwicklung, in der ab Mitte der 1970er Jahre immer weitere Funktionen in MRP-Systeme integriert wurden, was wiederum zu einer Ausdifferenzierung in ERP, CRM und SCM führte (Peeters 2009, 57). Weiter zurückreichende konzeptuelle Verästelungen ruft Wilson auf, indem er die historischen Bezüge zwischen Material Requirements Planning der 1960er und dem Konzept der Planning Office von Frederick W. Taylor von 1903 darstellt. Auch wenn Taylors Gesamtkonzept nicht unmittelbar umsetzbar gewesen sei und überkomplex, seien trotzdem Teilsysteme und die konzeptuelle Ausarbeitungen Taylors in das MRP der 1960er Jahre eingegangen. »Das Planungsbüro scheiterte aus vielen Gründen, aber ein wesentlicher Faktor war sein starker Bedarf an manueller Informationsverarbeitung« (Wilson/James M. 2016).<sup>cxlvi</sup> Die Kontinuitäten von der manuellen Informationsprozessierung über die frühen Lochkartensysteme zur computerisierten Informationsverarbeitung insbesondere bei den Bills of Material seien unübersehbar, auch wenn dies bisher unterhalb der öffentlichen Wahrnehmung liegt (ebd.). Eva-Maria Nyckel argumentiert am Beispiel der Software Salesforce, das heutige ERP-Systeme Taylors Vorschläge in Form zentralisierter Plattformen realisieren (Nyckel 2020, 26). Neben diesen konzeptuellen Rückbezügen gibt es zahlreiche genealogische Stränge, die sich in den Praxen und Materialien widerspiegeln, beispielsweise in der Verwendung von Lochkartentechnik, Formularen und Tabellen oder dem Delegieren in Transaktionen. Diese Verästelungen und Schwellen eröffnen neue Blicke auf die Kontinuitäten und Diskontinuitäten der Softwareproduktion von wirtschaftlichen Informations- und Prozessmanagementsystemen, die im Vergleich zu den Softwarepaketen und Apps für Endkunden, wie Microsoft Office oder Adobe Creative Suite in der Medientheorie bisher zu wenig Aufmerksamkeit erfahren haben.<sup>224 cxlvii</sup>

Die Entstehung der SOPS in der DDR ist ein Ergebnis von Zentralisierungs- und Rationalisierungsbestrebungen und war Folge eines Lernprozesses: Erst nach einigen Misserfolgen und Umwegen schälte sich ein modulares Konzept heraus. Im Abschnitt 3.2.1 *Dresden & Walldorf: Den Produktionsprozess koordinieren* wird ausgehend von einer Software zur Stücklistenverwaltung in der DDR aufgezeigt, wie die koordinierenden Praxen in Softwarefigurationen, aus denen die frühen Datenbanktechnologien hervorgingen, funktionierten. Die Abschnitte 3.2.2 *Dresden, Berlin(Ost): Digitalisierung und Vereinheitlichung*, 3.2.3 *Dresden, Guben, Moskau, Rostock: Sachgebietsorientierte Programmiersysteme (SOPS)* und 3.2.4 *Walldorf: SAP und die modulare Integration* erörtern Materialitäten und Praxen, und der Abschnitt 3.2.5 *Dresden, Walldorf: Konzepte für die Formalisierung, Formatisierung und Informatisierung der Realität* destilliert auf dieser Grundlage die grundlegenden Konzepte, welche sich aus den frühen

<sup>223</sup> Die Verwendung des westlichen Begriffs Material Requirements Planning kann als normative Setzung erscheinen, welche den »westlichen« Begriff als Standard setzt. Der Autor strebt trotzdem an, SOPS zuvorderst durch den Blick in das Material selbst zu beschreiben, auch wenn MRP als Kontextualisierung aufgerufen werden.

<sup>224</sup> Rossiter verdeutlicht die Dringlichkeit dieser Fragestellung: »Angesichts des starken Einflusses und der weitverbreiteten Verwendung von SAP-Software in Betrieben, müssen wir fragen, warum diese Firma und ihre Produkte derart unbekannt bei Forschenden in den Feldern der Medienwissenschaften, der Digital Humanities, der Softwarestudien und der Netzwerkkulturen sind« (Rossiter 2016, 52). Einer der Gründe dafür liegt in der vergleichsweise schwierigen Zugänglichkeit von ERP-Software. Im Gegensatz zu Textbearbeitungssoftware, die auf jedem PC per App-Store verfügbar ist, kostet aktuelle ERP Software als Service mehrere Tausend bis Zehntausende Euro in der Grundausstattung pro Monat. Ihre Konfiguration ist nur im Zuge intensiver Auseinandersetzung, z.B. durch Schulungen zu erschließen.

MRP in die heute gebräuchlichen ERP, CRM, und SCM hindurchziehen: Automatisierte Maintenance, modulare Integration, die Informatisierung von Realität und den DDR-Diskurs um Rationalisierung. Dies wird sowohl für Ost, also auch West geleistet und so werden neben den SOPS auch die ›westlichen‹ Softwarefigurationen COPICS, MAPICS und SAP historisch beleuchtet. Dabei wird von der These ausgegangen, dass MRP-Figurationen einerseits Datenbanken in ihrer heutigen Form überhaupt erst hervorgebracht haben, und andererseits Enterprise Resource Planning, Customer Relationship Management und Supply Chain Management in ihrer heutigen Form ohne Datenbanken nicht existieren würden.

In diesem Zuge wird deutlich werden, dass diese Figurationen aus emergenten bürokratischen, verwaltenden, also koordinierenden Praktiken entstammen, und dass kybernetische Überlegungen von Feedback und Kontrolle, eine geringere Rolle spielen, als gemeinhin angenommen wird. Gleichzeitig stellt das vorliegende Kapitel den nicht-relationalen Vorlauf dar, in welchem jene Bedürfnisse und Anforderungen an Datenbanksoftware entstanden, die sich in den relationalen Datenbankmanagementsystemen realisierten. Auf welche Weise, neben dem Einsatz westlicher Software, die in der DDR eigenentwickelten SOPS und DABA 1600 der Mehrfachnutzung, Standardisierung und Skalierung von Informationsprozessen in der DDR dienen, soll im Folgenden ausgeführt werden.

Die langen Bögen der Prozessmanagement-Figurationen reichen im Westen von den 1970er Jahren bis in die Gegenwart. Spürbar werden sie auf dramatische Weise, wenn die alternden Software-Figurationen, die häufig durch zahlreiche Ergänzungen schwer pflegbar geworden sind, durch neue Software ersetzt werden soll. Zu hohe Komplexität führte beispielsweise 2012 zum Scheitern des ERP-Projektes *Passion for Performance* beim Versandkaufhaus Otto welches ein System aus den 1980er Jahren, das sukzessive erweitert worden war, ablösen sollte. In der Folge entschied sich das Management für eine dezentrale Lösung, die einige vorhandene Teile re-integrierte (Jensen/Schwarzer 2013). 2015 baute die Deutsche Post das neu geplante *New Forwarding System* zum 30 Jahre alten *Logis* zurück, und schrieb in der Folge 508 Millionen Euro ab (Goebel 2015). *Logis* ist weiterhin beim Logistiker DHL im Einsatz. Das ERP-Projekt *Elwis* brach der Handelskonzern Lidl 2018 nach sieben Jahren ab, mit Verlusten von schätzungsweise 500 Millionen Euro (Kannenberg 2018). Im Jahr 2019 wurde in der Frankfurter Allgemeinen Zeitung über die mittelständische Firma Liqui Moly berichtet, welche von einem älteren, stark angepassten System auf ein aktuelles, modulares ERP-System umsteigen wollte, damit schwerwiegende Probleme aufgrund nicht erreichter Funktionalität erfuhr, und in der Folge mit Gewinneinbußen von 30% kämpfte (Preuß/Stuttgart 2019).

Vor diesem Hintergrund lohnt sich ein Rückblick auf die Genese von Prozessmanagementsoftware, denn es lässt sich deren Eingebundensein und Persistenz aufzeigen – zumindest im Westen, der keiner schlagartigen De-Industrialisierung unterlag, wie die Folgelandschaften der DDR. Da der Focus des vorliegenden Textes auf Verwendungsweisen relationaler Datenbankmanagementsysteme liegt, gilt es zudem, deren Genese im Verhältnis zu Figurationen des Prozessmanagement und des Informationsmanagement zu beobachten. Machte die relationale Datenhaltung im Vergleich zu anderen Informationsmodellen einen Unterschied?

Während die vorhergehenden Abschnitte vorrangig aus der östlichen Perspektive berichtet haben, changiert der Fokus nun zwischen Osten und Westen. Im Laufe des Abschnittes wird dabei eine zunehmende Verquickung zu bemerken sein, welche Gemeinsamkeiten und Unterschiede zwischen Ost und West der 1970er–1980er Jahre herausarbeitet. Einen Einstieg in diese Genealogie bietet der Versuch des ehemaligen Robotron Mitarbeiters Rolf Gräßler, Parallelen zwischen der Figuration der SOPS und dem modularen ERP der Firma SAP zu ziehen. In einem Artikel in der Zeitschrift *Die Welt*, erschienen im Oktober 2007, fasst der Journalist Lutz Frühbrodt zusammen: »Zwischen 1967 und 1975 arbeiteten

rund 300 Robotron-Mitarbeiter im Dresdner Stammhaus an den SOPS. Laut Gräßler habe man für die Entwicklung der DDR-Betriebssoftware 1635 Arbeitskräftejahre benötigt« (Frühbrodt 2007). Im Anschluss ruft er die andere Seite des Ost-West-Konflikts und den zum Mythos gewordenen Gründungsakt der Firma SAP auf: »SAP (Systemanalyse und Programmentwicklung) wurde 1972 von nur fünf ehemaligen IBM-Managern gegründet, die dann binnen kurzer Zeit erste Software-Produkte auf den Markt brachten« (Frühbrodt 2007). Die Schilderung fährt damit fort, vergleichend beide Entwicklungen zu betrachten: Ein Jahr nach der Gründung veröffentlichten die fünf SAP-Gründer die Finanzbuchhaltung SAP RF. Ab 1978 stellten sie mit R/2 eine modulare Anwendung für Großrechner zur Verfügung die aus einzelnen Programmteilen bestand, welche per Kommandozeile, d.h. Programmaufruf und Paramtereingabe zu bedienen war, ebenso wie die SOPS.

Diese zeitliche Abfolge nahm in den späten 1990ern Rolf Gräßler zum Anlass, die übliche technikhistorische Historiographie geographisch umzukehren: Die SAP in Westdeutschland hätte u.a. aus den Veröffentlichungen in *rechentechnik/datenverarbeitung* von den ostdeutschen SOPS erfahren und das Konzept abgekupfert (Gräßler In: Frühbrodt 2007; vgl. auch Hartmann 2009). Gibt es dafür Anhaltspunkte?

### 3.2.1 Dresden & Walldorf: Den Produktionsprozess koordinieren

»Haben also ERP Systeme das Problem der Materialsteuerung in großen Produktionsbetrieben abschließend gelöst? Für das Ausmaß der Probleme, wie es die Hersteller in den 1950er und 1960er Jahren betraf, ist die Antwort ein eindeutiges Ja. Heute sind Vorlaufzeiten von mehr als einem Jahr oder sogar mehreren Monaten für professionelle Produkte inakzeptabel« (Peeters 2009, 66).<sup>cxlviii</sup>

Die Untersuchung der These, dass koordinierende, modularisierte und standardisierte Informations- und Prozessmanagementsysteme ohne das Konzept elektronischer Datenbanken nicht möglich gewesen wären, führt in Museen und Sammlungen früher Informationstechnik, in Zeitschriftensammlungen und Archive. Sie benötigt eine besondere Aufmerksamkeit für Tabellen und Formulare, aber auch für die Beschreibungen modularer Softwaresysteme und jener Veröffentlichungen, welche die Konzepte ›Prozessmanagementsystem‹ und ›Standard‹ zusammenführen. In Erweiterung der These soll gefragt werden, ob relationale Datenbanksysteme einen spezifischen Beitrag für modularisierte und standardisierte Informations- und Prozessmanagementsysteme leisteten. Bevor zu diesem Schritt angesetzt werden kann, sind die Kontexte nicht-relationaler Systeme der 1960er und 1970er Jahre näher zu beleuchten.

Dramatisch verlief die ›Rettung‹ eines DDR-Großrechners ESER EC 1056 durch den Verein Computermuseum Rechenwerk Halle. Aus den Ruinen des Wissenschaftlich-Technischen Zentrum der Landwirtschaft in Markkleeberg sicherten 2007 die Vereinsmitglieder den Großrechner, den sie auf einem 7,5-Tonnen-LKW abtransportierten: »Nachdem wir durch das leerstehende Verwaltungsgebäude und einen ziemlich maroden Verbindungsgang gelaufen sind, schloss uns die Dame [eine Mitarbeiterin des Liegenschaftsamtes, welches die Immobilie 2006 verwaltete – F.H.] die schwere Stahltür zum eigentlichen Rechnerraum auf – in welchem im Halbdunkel große blaue Würfel auszumachen waren. Seit 18 Jahren stand in diesem Raum ein Großrechner ESER EC1056, der vorletzte Typ des ESER.

Einige Komponenten wie einige Wechselplatten, der Prozessor und die Bedieneinheit fehlten allerdings. [...] Tags darauf gruben wir uns zu zweit durch die beiden Räume, in denen bis ca. 1m Höhe Magnetbänder lagen. [...] Das Gesamtgewicht der Technik liegt im Bereich von 4,7 t« (Rechenwerk Halle 2007).

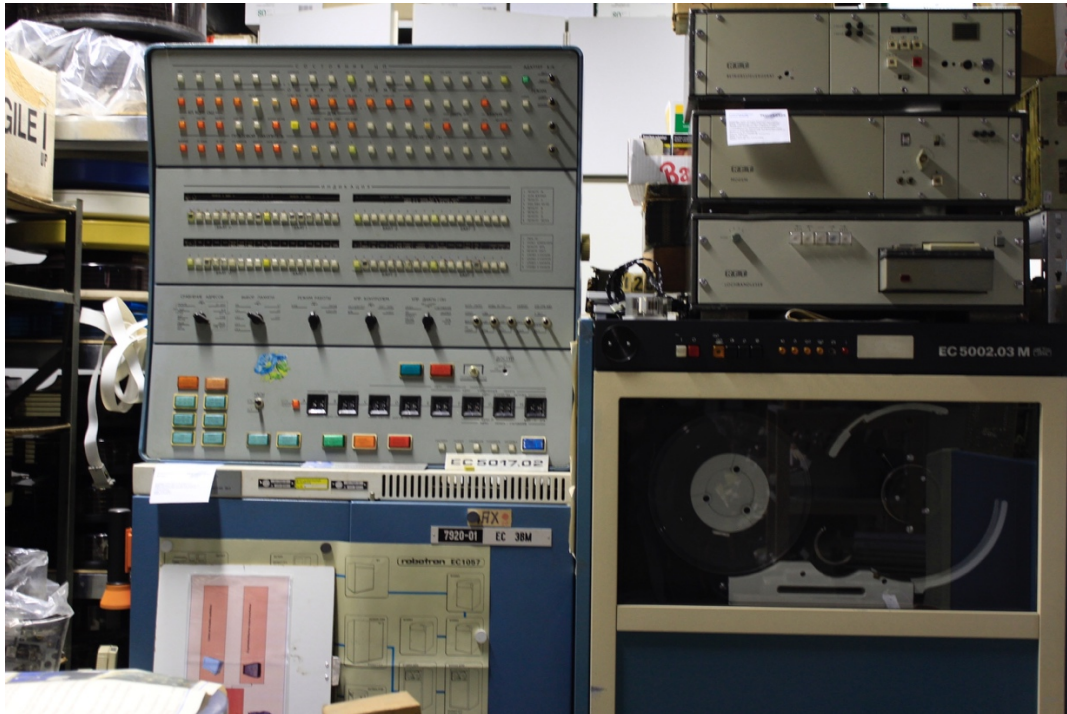


Abb. 52: ESER-Bedienkonsole und Bandlaufwerk in der Sammlung Rechenwerk Halle/Saale (Autor)

Heute stehen diese und weitere ESER-Bauteile in einer ehemaligen Kaufhalle in Halle/Saale, in der Saalfelder Str. 11, als Teil einer Sammlung von mehr als 2000 Computern und Rechenmaschinen, vorrangig aus der DDR (Abb. 52). Lauffähig ist der ESER EC 1056 momentan nicht. Die ehrenamtlichen Vereinsmitglieder bemühten sich um die Wiederherstellung des Festplattengerätes in der Hoffnung, die Daten zahlreicher Plattenstapel auslesen zu können, die sie bereits gesammelt hatten (Rechenwerk Halle 2011). Was genau sich auf den Festplattenstapeln und den nur teilweise beschrifteten Magnetbändern befindet, harret seiner medienarchäologischen Erforschung – tatsächlich verfügt der Verein Rechenwerk über das elektrotechnische Wissen, um Disketten, Festplatten, Magnet- und Lochbänder zu rekonstruieren. Es wäre keine große Überraschung, wenn sich darunter neben dem Betriebssystem und den Anwendungsprogrammen auch die Datenbanksoftware *Bankspeichersystem technischer Information* (BASTEI) oder Datenbanksystem/Robotron (DBS/R) befinden würde, die Teil eines größeren Software-Komplexes, der *Sachgebietsorientierten Programmiersysteme* (SOPS) war.<sup>225</sup>

Würde es gelingen, auf diese Daten und Dateien zuzugreifen, gälte ein erster Handgriff den Bediendbüchern *Problembeschreibung – Sachgebietsorientiertes Programmiersystem BASTEI* von 1972 und *Systemunterlagen-Dokumentation* von 1974, die durch den VEB Robotron herausgegeben wurden. In Ermangelung eines tatsächlich funktionierenden historischen ESER-Systems mit BASTEI, soll im

<sup>225</sup> Tatsächlich verfügt der Verein über einen bootfähigen DDR 16-bit Rechner K 1600, auf welchem das relationale DBMS DABA 1600 lauffähig wäre. Allerdings ist derzeit keine lauffähige Kopie der Datenbanksoftware mehr vorhanden.

Folgenden Abschnitt die grundsätzliche Vorgehensweise aus den in der Universitätsbibliothek Dresden vorliegenden Handbüchern rekonstruiert werden.

Die Vorgehensweise in BASTEI entspricht dem westlichen Bills Of Materials Processor (BOMP) von IBM, welcher konzeptuelle Vorlage für die östliche BASTEI war, aber auch für CINCOMS erweitertes westliches Datenbanksystem TOTAL DBMS (McGee 1981, 507). BOMP war durch IBM-Programmierer entwickelt worden, um die Stücklisten zu verwalten. Es kam zuerst bei Gerätetechnik-Firmen wie John Deere oder dem Gerätebauer Allis-Chalmers zum Einsatz – Firmen, deren Produkte aus zahlreichen Einzelteilen und Baugruppen bestanden.

»Die Entwicklung von BASTEI umfasste ein Neuschreiben des Programmcodes, die Anpassung an die verfügbaren ESER-Betriebssysteme sowie eine Reihe funktionaler Erweiterungen zur Verbesserung der Dateneingabe sowie der Auswertbarkeit. Die Prinzip-Struktur wurde vom Vorbild übernommen« (Bittner 2008, 270). Dafür lag der Quellcode von BOMP vor, welcher im Sinne einer Adaption »durch Ersetzen der Namen aller Programmelemente wie Datenobjekte, Variablen, Konstanten« (Bittner, in: Interview Bittner/Heinemann 2018, N. 3) für die Verwendung auf DDR-Computern angepasst wurde. BASTEI wurde Teil der innovativen, modularen Sachgebietsorientierten Programmiersysteme (SOPS), die durch Robotron in der DDR für Großrechner adaptiert und entwickelt wurden. Hier kündigen sich bereits die vielfältigen genealogischen Verflechtungen an, denen im Folgenden nachzugehen ist.

Wie konnte eine ›kapitalistische‹ Software Teil eines ›sozialistischen‹ Standardsystems betrieblicher Software werden? Finden sich Spuren einer Erklärung in der Bedienungsanleitung, oder gibt es andere Dokumente, die darüber Auskunft geben können? Wie sind BASTEI und andere Programme der Sachgebietsorientierten Programmiersysteme (SOPS) in die Softwarelandschaft ihrer Zeit einzuordnen? Wie funktionierte dieses Datenbanksystem überhaupt und unter welchen Prämissen wurde es ursprünglich entwickelt? Und schließlich: Wie gestaltete sich die Nutzung von BASTEI/BOMP, welcher Bedarf ging ihr voraus?

Bevor wir uns diesen Fragen zuwenden können, gilt es, ein grobes Verständnis für die Funktionsweise von BASTEI/BOMP zu erarbeiten, insofern es sich aus den Bedienungshandbüchern rekonstruieren lässt.

### 3.2.1.1 Explosionen: Bankspeicherung Technischer Informationen (BASTEI) / Bill Of Materials Processor (BOMP)

Die Vorgehensweise in den Handbüchern<sup>226</sup> war wie folgt beschrieben: Es gibt vier Dateien:

- Die ›Gegenstandsdatei‹ enthielt alle Rohmaterialien, Einzelteile oder Baugruppen und deren Eigenschaften (z.B. Artikelnummer, Farbe, Benennung).
- Die ›Stücklistendatei‹ stellte die Einzelteile dar, die für ein Stück, d.h. ein Produkt, benötigt werden.
- Die ›Arbeitsplatzdatei‹ enthält alle verfügbaren Maschinen- oder Handarbeitsplätze sowie deren Eigenschaften (z.B. Bezeichnung, Kostenstellen, Verarbeitungskapazität).
- Die ›Arbeitsplandatei‹, gibt die Reihenfolge der einzelnen Arbeitsschritte, um komplexe Produkte herzustellen, vor.

---

<sup>226</sup> Vgl. *Problembeschreibung – Sachgebietsorientiertes Programmiersystem BASTEI (1972)* und *Systemunterlagen-Dokumentation – Sachgebietsorientiertes Programmiersystem BASTEI (1974)*.



Diese vier Hauptdateien waren über Verbindungsdateien verknüpft, welche Paare von Speicheradressen enthielten, sogenannte Adressverkettungen, mit denen zwei Adressen miteinander verkettet wurden.<sup>227</sup>

Bei BASTEI handelte es sich nicht um ein einziges Programm, sondern um eine Sammlung von miteinander interagierenden Komponenten. Die in BASTEI verfügbaren Teilprogramme lassen sich grob unterscheiden in ein Leitprogramm, welches das Laden und Speichern der Dateien steuerte (LEITPROG), ein Modul zur Dateiorganisation (DATEIORG), Programme zur Eingabe und Ausgabe der Daten auf den verschiedenen Geräten (EAWURZEL, EAARBEIT, EAEROBAS), ein Modul zur Reorganisation der Dateien (BDAREORG) und ein Modul, welches die Verbindungsdateien zwischen den vier genannten Hauptdateien reorganisieren konnte (VDAREORG). Diese Programme griffen ineinander und wurden durch LEITPROG aufgerufen und abgearbeitet. Die Arbeit mit BASTEI war aus heutiger Sicht kompliziert und fehleranfällig. Zuerst mussten die benötigten Programme zusammengestellt werden und von Lochband, Lochkartenstapel oder Magnetband in den Hauptspeicher eingelesen werden (LEITPROG BA#B02). Dann konnten sie nacheinander ausgeführt werden und Daten in den Speicher einlesen und verarbeiten (DATEIORG BA#B06) und ausgeben (BA#B08). All dies geschah, indem direkt in Speicheradressen gelesen oder geschrieben wurde (z.B. die Gegenstandsdatei mit EXTENT SYS004, BASTEI, 1,2,970,40. Dabei steht 970 für die Speicheradresse und 40 für die Zeichenlänge, die hinzugefügt wird.). Der Speicher war in verschiedene, feste Teilbereiche unterteilt, welche das Ablaufprogramm enthielten, und davon getrennt jeweils Gegenstands-, Stücklisten-, Arbeitsplatz- und Arbeitsplandatei in eigenen Speichersektoren. Als Zwischenspeicher für Sortiervorgänge oder die Vorbereitung von Druckdaten waren weitere Speicherbereiche zu reservieren, gegebenenfalls auch auf Magnetband oder Lochkartenstapel. Es wird deutlich, dass bei einer derartigen Vorgehensweise das Überschreiben bestimmter Speicherbereiche unbedingt vermieden werden musste, und dass die Programmierer, die gleichzeitig auch die Nutzer der Datenbank darstellten, die Speicher-Adressverwaltung mit großem Aufwand selbst betreiben mussten.

Eine Rückmeldung zum Programmablauf erhielten sie über Ausdrücke, die kryptische Fehlermeldungen ausgaben. Die Meldung »BG baa068i / BG beispad lad / BG 12 / BG 427« bedeutet beispielsweise, dass beim Laden des Arbeitsplanes (BG baa068i) für Datensatz 12, der zugehörige Datensatz zum Arbeitsplatz 427 nicht eingelesen werden konnte. Gleichwohl war BASTEI gegenüber den meisten vorhergehenden Datenverwaltungssystemen ein Fortschritt, denn es lieferte viele benötigte Routinen bereits integriert in das Gesamtkonzept mit, und diese mussten nicht mühsam durch die Programmierer selbst erstellt werden. Der besagte fehlende Arbeitsplatz 427 konnte beispielsweise im Nachgang mit Hilfe des Programmes »Hinzufügen« in die gespeicherten Daten per Lochkarte aufgenommen werden.

In vielen Eigenentwicklungen von Datenverwaltungssystemen waren derartige Operationen nicht oder noch schwieriger möglich. Das Programm mitsamt des Dateneinlesens musste komplett neu geladen werden – und die Programmierer hofften, dass nicht an anderer Stelle ein Fehler auftrat. Ähnliches galt für die Operationen »Berichtigen«, »Löschen« und »Stillegen«, welche unweigerlich die Reorganisation der betroffenen, abhängigen Datensätze und damit die Änderungen der Dateien, in denen sich diese Datensätze befanden, nach sich zog. Um die Komplexität der Unternehmung zu verdeutlichen, gilt es sich zu vergegenwärtigen, dass alle Daten in den Dateien sortiert vorlagen, und dass die

---

<sup>227</sup> Wie diese Adressverknüpfung im Falle von BOMP funktionierte, zeichnet *Early MRP Systems at Royal Philips Electronics in the 1960s and 1970s* (Peeters 2009, 60) nach.

Sortierreihenfolge bei allen Lese und Schreiboperationen mitzudenken war.<sup>228</sup> BASTEI war darauf vorbereitet, durch andere Anwendungen als Datenspeicher verwendet zu werden. Diese anderen vorgefertigten Anwendungen wurden unter dem Begriff Sachgebietsorientierten Programmiersysteme zu einem Programmpaket verpackt (1972; 1974). Mit diesem grundsätzlichen Verständnis von BASTEI/BOMP ist es nun möglich, darzustellen, wie diese frühe Datenbanktechnologie und -praxis in die SOPS integriert wurde.

### 3.2.1.2 Automatisierte Maintenance mit COPICS und MAPICS

Als ehemalige IBM-Programmierer in Mannheim kannten die SAP-Gründer mit BOMP zumindest die gleiche Software, wie sie auch den Robotronern bekannt war. Die IBM Systeme *Production Information Control System* (PICS) und *Communication Oriented Production Information Control System* COPICS, welches teilweise bei IBM in München entwickelt wurde, dürften ihnen bekannt gewesen sein. Es lohnt sich daher, diese Industrie-Softwaresysteme näher zu beschreiben, um Gräßlers Idee des Technologietransfers von Osten nach Westen auf den Grund zu gehen. Dieser Frage soll eine weitere beigelegt werden. In welchem Verhältnis stand die Genese des Datenbank-Konzepts zu Enterprise Resource Planning (ERP), Customer Relationship Management (CRM) und Supply Chain Management (SCM) Softwaresystemen?

Setzen wir den anfänglichen Focus also auf IBM BOMP, da sich darin die Expertise von Robotron und SAP überschneiden. Daher gilt es, die Tiefenschichten der Entstehung von BOMP aufzusuchen und diese vorübergehend für eine Betrachtung freizulegen. IBMs Bill Of Materials Processor (BOMP) war als Stapelverarbeitungssystem für eine Reihe von Typenprojekten (um in Gräßlers Terminologie zu bleiben) entwickelt worden. Anfang der 1960er Jahre erstellte in den USA der IBM-Programmierer Gene Thomas gemeinsam mit Joseph Orlicky, einem Mitarbeiter des Traktorherstellers J.I. Case Corp. in Racine, Wisconsin eine Stücklistenverwaltung, die später bei weiteren Landwirtschaftsgeräteherstellern wie John Deere, Allis-Chalmers zum Einsatz kam (Henslee 2016).<sup>229</sup> Die Entwicklung erfolgte auf dem IBM 305 Ramac, dem ersten Großcomputer mit Random-Access-Zugriff durch Festplattenspeicher. Laut Thomas bedurfte es zwei Jahre konstanter Diskussionen innerhalb IBMs und der Fürsprache der bereits vorhandenen Industrienutzer, bis BOMP als eigenes Produkt durch IBM anerkannt wurde. »Jeder war der Meinung, dass die Ingenieursleistung viel zu komplex wäre, um über Produktlinien [verschiedener] Branchen hinweg standardisierbar zu sein« (Thomas 2008, Abschn 16).<sup>cxlix</sup> Ende der 1960er Jahre dann erkannten die IBM-Entwickler anhand ihrer Arbeit für den Traktorenhersteller J.I. Case Corp, dass BOMP nicht nur zur Stücklistenverwaltung, sondern genereller noch als »File Organisation Program«, bzw. Datenbank verwendet werden könne (ebd.).

Historisch bedeutet dies einen Medienwechsel, von einem spezifischen Anwendungsprogramm zur generalisierenden Datenbank. Dieser Medienwechsel findet im Rahmen der gleichen Softwarefiguration statt, die nun jedoch eine andere Nutzung erfährt. Diese andere Nutzung ist gekennzeichnet durch eine

---

<sup>228</sup> Diese Beschreibung fasst zusammen, welches die Hauptprobleme dieser Form der Datenorganisation waren, die in den 1980er Jahren durch die relationale Algebra gelöst werden sollten: Adressierung, Sortierung, Speicherzuweisung waren im netzwerk-orientierten BASTEI durch die Programmierer zu verwalten und mit hohem Aufwand verbunden.

<sup>229</sup> Die Blogbeiträge von *From BOMP to SaaS and Beyond – 1960s: BOMP & the Early Days of MRP* (Henslee 2016) und *Eras of MRP to LEAN* (Thomas 2008) geben einen Überblick und Verweise auf Seiten weiterer Industrieveteranen, die aus ihrer Perspektive über die noch wenig beleuchtete Früh- und Vorgeschichte der ERP berichten. Detailliert siehe auch *Early MRP Systems at Royal Philips Electronics in the 1960s and 1970s* (Peeters 2009, 59).

Änderung auf semantischer Ebene, die nicht mehr die Beziehungen spezifischer Domänen (Gegenstand, Stücklisten, Arbeitsplatz, Arbeitsplan) im Blick behält, sondern die Abbildung von Beziehungen jeglicher Informations-Domänen.

Im Osten machte man die gleiche Entdeckung: Genau diese andere Verwendung als Datenbank nach der Netzwerkmetapher erfuhr BOMP als Adaption mit dem Namen BASTEI in der DDR. So eröffnet die BASTEI-Dokumentation vom 30.5.1972 nicht mit dem Hinweis auf die Stücklistenproblematik, sondern zeigt viel breitere Möglichkeiten an: »Das Programmiersystem BASTEI bietet die Möglichkeit, eine Datenbank für ein *beliebiges Anwendungsgebiet* zu betreiben« (1972, 7, Hervorh. F.H.).

BOMP/BASTEI markiert damit im Westen und im Osten einen der Anfänge der Übergangsphase von der zeitverzögerten Stapelverarbeitung zur Echtzeit-Anwendung. Erst die Idee der generalisierten Datenbank in Verbindung mit einem schnellen Geräte-Input/Output ermöglicht den Daten-Zugriff in Echtzeit. Um den schnelleren Input/Output zu ermöglichen, können die Daten nicht mehr sequentiell von Lochkarten stapelweise verarbeitet werden. Vielmehr müssen sie in Direktzugriffsspeichern, entweder dem Hauptspeicher oder einer Festplatte mit geringer Zugriffszeit, abgelegt werden. Dies erforderte einen Wechsel der Materialität.

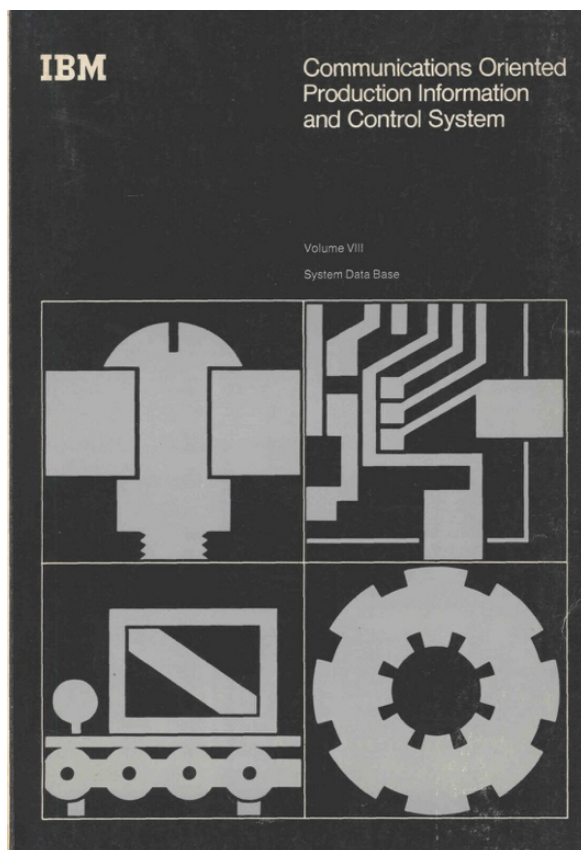


Abb. 53: *System Data Base* – Cover der IBM Broschüre *Communication-Oriented Production Information Control System*, Band 8 (IBM 1972a).

Im Westen konnten die Ingenieure auf neue Gerätegenerationen mit Direktzugriffsspeicher eine neue Softwaregeneration aufbauen: Aus der Programm- und Informationssammlung *Labor & Material Planning* (LAMP)<sup>230</sup> von 1964 wurden seit 1966 die Produktionsmanagementsysteme *Production*

<sup>230</sup> Auf IBM-Kleincomputern, z.B. dem System 36 folgten 1977 mit dem Manufacturing, Accounting and Production Information Control System (MAPICS) eine kleinere Variante des Großcomputer-COPICS (IBM (Hrsg.) 1983; Taylor/Norm 1987). Zahlreiche weitere ähnliche Programmpakete wurden durch andere Hersteller angeboten (Thomas 2008, Abschn 20).

Information Control System (PICS), und seit 1972 das *Communication-Oriented Production Information Control System* (COPICS) entwickelt (Thomas 2008, Abschn 18).

COPICS (Abb. 53) war ein Best-Practice-Sammlung, die durch IBM herausgegeben wurde und die Verwaltung zahlreicher betrieblicher Prozesse betraf: Konstruktions- und Produktionsdaten, Kundenmanagement, Produktionsvorhersage, Materialverwaltung, Fertigungssteuerung et cetera. »Die COPICS-Broschüren wurden am Manufacturing Industry Center der IBM München von einer Expertengruppe geschrieben, welche die PICS-Entwickler Thomas Glaza und Don Ralston mit kleineren Beiträgen von [Joseph] Orlicky<sup>231</sup> und [Oliver] Wight umfasste, und ohne Namensnennung publiziert wurde« (Peeters 2009, 60).<sup>d</sup>

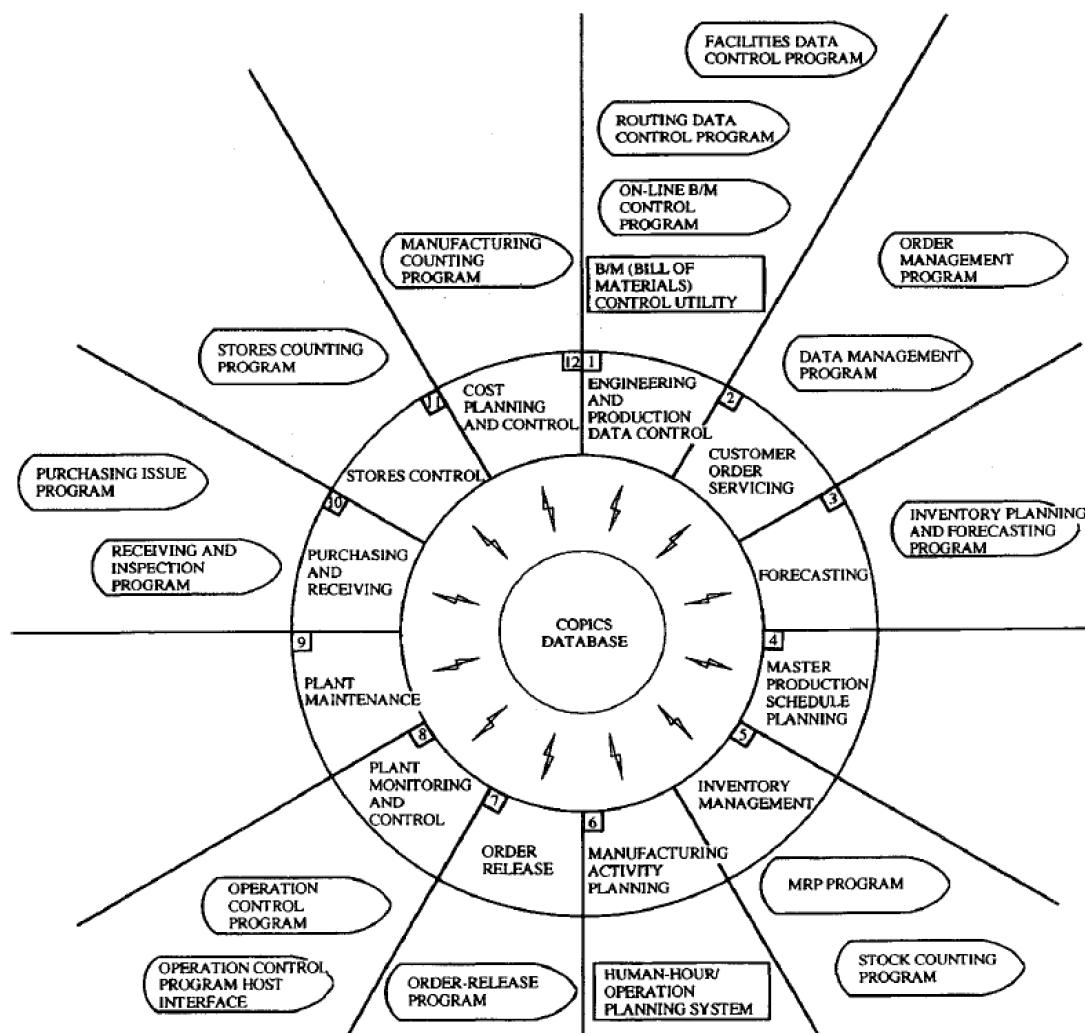


Abb. 54: IBM COPICS Software Module, aus IBM: *Communication Oriented Production Information Control System* (1972) Vols. I—VIII, zitiert nach (Hitomi 1996, 447).

Das System aus 12 Modulen konzentrierte sich auf Fragen der Produktion und beinhaltet keine Komponenten zu Vertrieb, Personalwesen und Buchhaltung (Abb. 54). Es kann daher als prototypisches Material Requirements Planning (MRP)-System gekennzeichnet werden, aus denen erst später

<sup>231</sup> Orlicky schrieb den Text über Datenbanken in Band 8.

Enterprise Resource Planning (ERP), Customer Relationship Management (CRM) und Supply Chain Management (SCM) hervorgingen. COPICS verfügte neben den Best-Practice-Büchern über einzelne Softwarekomponenten, die ab 1982 in einer der zweiten Versionen miteinander integriert wurden. Sie verwaltete die Daten ab 1982 in der hierarchischen IBM-Datenbank Information Management System/DLI (IMS) und mit dem Transaktionsmonitor IBM Customer Information Control System (CICS).<sup>232</sup> IMS und CICS bilden somit auf Datenbankseite den Umbruch von Stapel zu Echtzeitverarbeitung ab. Später fand in COPICS auch IBMs relationales Datenbanksystem DB2 Verwendung, welches ab 1983 verfügbar war.

Der an COPICS beteiligte IBM-Ingenieur Joseph Orlicky veröffentlichte 1975 das Buch *Material Requirements Planning*, welches übliche Computer-basierte Praktiken des »Business Information Processing« zusammenfassen und neu etablieren sollte, und das zum Standardwerk seiner Zeit avancierte.<sup>233</sup> Der fünfte Teil des Buches widmet sich dem Umgang mit Daten und definiert dort ein MRP-System als eine Menge logisch miteinander verknüpfter Inventar-Datensätze, welche durch ein Programm verknüpft seien, das diese Datensätze aktuell halte. Die Veränderungen an Werkstücken oder Einzelteilen, z.B. deren Entnahme aus dem Lager und Bereitstellung an der jeweils benötigten Maschine bezeichnet Orlicky als Transaktionen, welche dynamische Transaktionsdaten generieren, im Unterschied zu statischen Daten, die den Objektcharakter der Einzelteile adressieren (Orlicky 1974, 173, 183). Der angemessene Verwaltungsort sei die »data base«, denn »Dateien konstituieren das Fundament auf welchem die Überstruktur der Softwareanwendung gebaut wird. [...] Das reibungslose Funktionieren und die Effektivität eines MRP-Systems sind, in erheblichem Ausmaß eine Funktion der *Qualität der Systemdateien*« (ebd. 189, Hervorh. im Original).<sup>ci</sup> Orlicky liefert damit einen Hinweis darauf, wie Daten medien-theoretisch kategorisiert werden können: Einerseits als eher statischen Stammdaten und andererseits als vergleichsweise dynamische Nutzungs- bzw. Transaktionsdaten.

Üblicherweise, so Orlicky weiter, seien die Datensätze in Firmen in einem schlechten Zustand, da die erforderliche Informationsarbeit, die Maintenance, in Firmen häufig geringgeschätzt werde. Auch die Kulturtechnik der periodischen Bereinigung von Datenbeständen sei unzureichend, denn »etwas in Ordnung zu bringen ist die eine Seite der Medaille, es aber konstant in Ordnung zu halten, ist die andere« (ebd. 190).<sup>clii</sup> Die manuelle Bestandsführung, so Orlicky sei grundsätzlich zu schwierig, als dass sie erfolgreich sein können. Fehler in den Datenbeständen würden allein durch die menschliche Fähigkeit zur Improvisation ausgeglichen werden können. Computerhersteller hätten daher massiv in entsprechende Software investiert, welche als »Werkzeug« (ebd.) die Datenorganisation unterstützen könnten. Orlicky verspricht hier eine Lösung, welche das In-Ordnung-Halten, die Maintenance, automatisiert und personalintensive Informationsarbeit automatisiert.<sup>234</sup> Teil dieser Lösung sollte es sein, Datenduplikate, -redundanz und verschiedene Versionen gleicher Datensätze weitestgehend zu reduzieren, sowie den Zugriff auf Daten zu optimieren. »Diese Ziele widersprechen sich« (ebd. 192),<sup>cliii</sup> denn Daten, die nicht als Duplikat existierten, seien immer nur für einen einzigen Vorgang gleichzeitig

---

<sup>232</sup> Welche Datenverwaltung COPICS in der ersten Phase zwischen 1972 und 1982 verwendete ist aufgrund der heutigen Unzugänglichkeit der Original-Dokumentation nicht mehr rekonstruierbar. Zur Bedeutung des Transaktionsmanagement CICS siehe Abschnitt 2.3.6 *On-Line Transaction Processing (OLTP)*.

<sup>233</sup> Bei dem in der Bibliothek der Universität Leipzig existierenden Exemplar handelt es sich um eine gebundene Foto-Kopie des Buches, welches Stempel der Sektion Mathematik der Uni Leipzig trägt mit dem Hinweis »Sektionsinternes Material für Lehre und Forschung«, der auf den Gebrauch in der DDR verweist.

<sup>234</sup> Zum Begriff der Infrastruktur, Maintenance und Reparatur siehe *Medium Infrastruktur – Trajektorien soziotechnischer Netzwerke in der ANT* (Schabacher 2013, 145f.). Das automatisierte In-Ordnung-Halten impliziert auch eine Reduzierung der Care-Tätigkeit der bisherigen Operateure und macht deren Leistung noch unsichtbarer als bereits zuvor.

verfügbar. Eine – wenn auch nicht vollständige – Lösung dieses Zielkonfliktes sah Orlicky in der Verwendung von Datenbanksystemen, welche Daten zentral organisierten. Dies werde jedoch zusätzlich verkompliziert, da die Daten nun nicht mehr zu einzelnen Betriebsabteilungen gehörten, sondern miteinander geteilt würden. Sie dienten nun vielmehr als eine »gemeinsame Basis« (ebd. 203, Hervorh. durch Orlicky),<sup>cliv</sup> auf welche die verschiedenen Computerprogramme zugreifen können. Dies sei aber nur möglich, wenn die mit der Datenerhebung befassten Mitarbeiter gut geschult seien: »Ebenso wichtig sind die technischen Aspekte bei der Gewährung der Datenintegrität – oder vielmehr noch – sind es Training, Disziplin, und die Motivation der Leute. ... [Sie] müssen von dem Fakt überzeugt werden, dass die Ausgabe eines Computers nicht besser sein kann als die Eingabe« (ebd.).<sup>clv</sup>

Wie wurden diese Wissenssammlung mit Softwarehinterlegung in der Praxis im Westen eingesetzt? Der Journalist Rory Baxter schilderte im britischen Fachmagazin *The Production Engineer* 1984 den Einsatz von COPICS beim Maschinenbauer KTM in Brighton als Erfolgsgeschichte unter dem Titel *How COPICS helped KTM grow in ›a harsh environment‹*. »Bevor dort ein computerbasiertes Produktionsplanungssystem installiert wurde, gab es in der Abteilung für Produktionsplanung dicke Luft und ständigen Streit. Das System, welches im Zentrum der ständigen Unstimmigkeiten stand, war ein ›typisches 10-jähriges System, das sich an großen Stapeln orientierte und eine Menge Papierkörbe füllte.« (Baxter 1984, 32).<sup>clvi</sup> Die neue Figuration bestand aus Softwarekomponenten, sechs Druckern und zahlreiche Terminals, die an einen Großrechner angeschlossen und über verschiedenste Abteilungen verteilt waren.<sup>235</sup> Eine externe Beratungsfirma führte nicht nur die Software ein, sondern wurde, so Baxter, zum »Katalysator« und »unpolitischen«, überparteiischen Berater indem sie als »Koordinator, Ausbilder und Systemdesigner« die organisatorischen Strukturen des Betriebes für den Einsatz von COPICS änderte. »Es ist offensichtlich üblich für eine Firma, welche ein spezifisches System in Einsatz hat, dass sie unfähig ist, sich eigenständig auf ein anderes Gleis zu setzen, denn die Mitarbeiter, die wissen, was getan werden sollte, können aufgrund der Firmenpolitik nicht mit den Managern sprechen« (ebd.),<sup>clvii</sup> gibt der Autor als Grund an. Der Artikel brachte die Einführung von COPICS auch in Zusammenhang mit der wirtschaftlichen Krisensituation Ende der 1970er Jahre in Großbritannien.<sup>236</sup> Die Einführung der Software habe das Überleben der Firma gesichert. Im Ergebnis dieser Umstrukturierung konnten Inventar und Umlaufbestand sowie die Lieferzeiten deutlich reduziert werden (ebd.).

Der Anspruch von COPICS in seiner zweiten Version war, im Unterschied zur Stapelverarbeitung in Echtzeit (engl. On-Line) Eingaben und Ausgaben zu ermöglichen (Hitomi 1996, 441–449).<sup>237</sup> Die Bedeutung von COPICS für die folgenden Softwarekonzepte verdeutlicht der Informatiker und Historiker Jos Peeters: »Die IBM-COPICS-Veröffentlichung von 1972 spielte eine Schlüsselrolle als offener Standard, welcher der allgemeinen Öffentlichkeit zur Verfügung stand. Von dieser Publikation ausgehend, kann eine Art genealogischer Baum gedacht werden, der zeigen würde, wie die Kenntnis eines Systems (und seiner Probleme) die Markteinführung eines anderen hervorrief, und so weiter« (Peeters 2009, 65).<sup>clviii</sup>

---

<sup>235</sup> Der Artikel diskutierte zwar ausführlich die Hardwarekomponenten, sagte aber wenig über die Softwarekonfiguration, die nur eine Teilkonfiguration des gesamten COPICS darstellte, aus.

<sup>236</sup> Großbritannien befand sich 1977 am Rande einer Staatspleite, die nur durch einen Milliardenkredit des IWF abgewendet werden konnte. Eine Inflationsrate von 17 bis 21 Prozent und zahlreiche aufeinander folgende Streiks gingen dem Wahlsieg Margret Thatchers 1979 voraus.

<sup>237</sup> Ob dieses Ziel erreicht wurde, bzw. ab welcher Programmversion lässt sich anhand der heute verfügbaren Literatur nicht nachvollziehen.

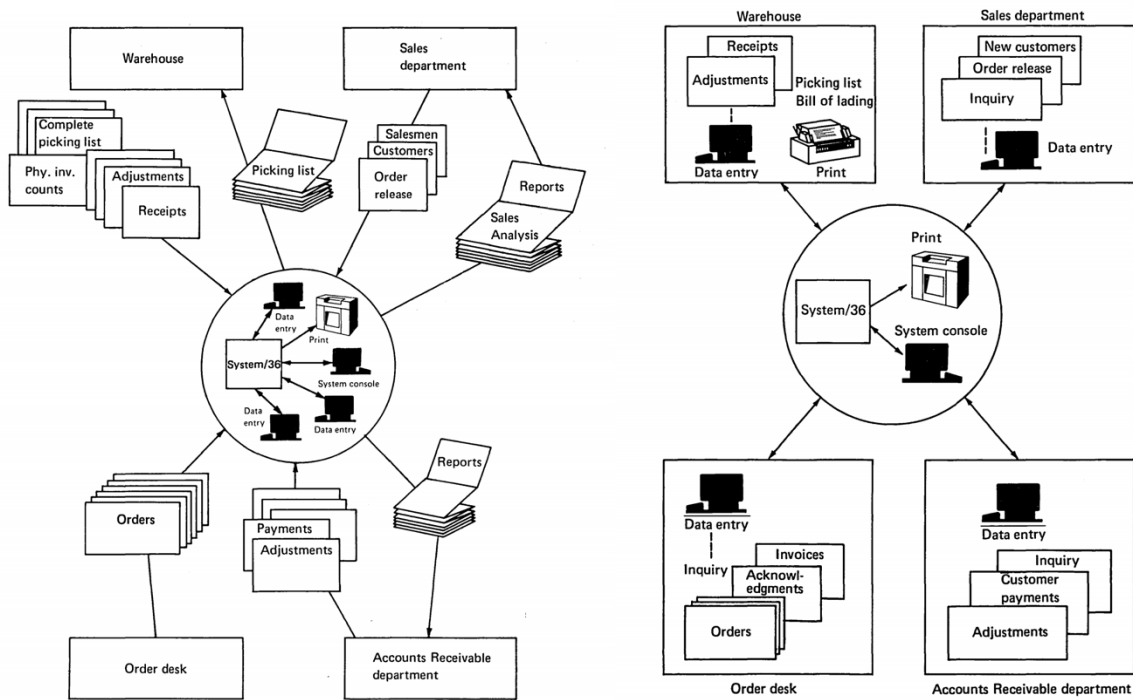


Abb. 55: In der Broschüre zu MAPICS (der kleineren Version von COPICS) werden zwei Installationsvarianten gezeigt: Links – zentralisierte Installation in einem Raum; Rechts – Dezentralisierter Dateneingabe und -ausgabe aus bis zu 1500 m Entfernung. Dies markiert den Übergang zu dezentralisierten MRP-Figurationen ab Mitte der 1980er Jahre (IBM (Hrsg.) 1983, Abschn 1–11, 1–12).

Der Schluss liegt nahe, dass die SAP-Gründer eher die acht IBM-Hefte zu COPICS und die Dokumentationen zu IBM MAPICS (Abb. 55), sowie die dazugehörigen Softwaremodule vor Augen hatten, als die Ausgabe der *rechentechnik/datenverarbeitung*, in der Gräßler das Konzept der DDR-Öffentlichkeit vorstellte. Beide, COPICS und SAP im kapitalistischen Westen, und SOPS im real-sozialistischen Osten, markieren ab 1972 die parallele Emergenz heutiger ERP, CRM und SCM Systeme. Diese Wahrnehmung setzt die von Gräßler entworfene Narration, und generell die Narrationen des Ost-West, bzw. West-Ost Technologietransfers unter Spannung. Es ist möglich, dass sich diese Spannung nicht in Richtung einer der beiden Seiten auflöst, wodurch die Historiographien der Ost-West-Technologien viel stärker als bisher in Schwebe gehalten werden können. Aus dieser Perspektive lohnt eine Erörterung, wo die Ähnlichkeiten der SOPS und der SAP-Software liegen. Dazu soll ein eingehender Blick in die Frühgeschichte der westdeutschen Firma SAP geworfen werden, um gegebenenfalls Vergleiche mit Robotrons SOPS-Konzept anstellen zu können.

### 3.2.2 Dresden, Berlin(Ost): Digitalisierung und Vereinheitlichung

#### 3.2.2.1 Pflichtenheft und Datenerhebung

Die Datenbank- und Verwaltungspraktiken um BASTEI/BOMP verzahnten sich mit einer Reihe von bereits existierenden Praxen des Prozessierens, anhand derer die Emergenzen medialer Figurationen der Kooperation genauer erläutert werden können. So war die praktische Einführung eines SOPS in mehrere Etappen unterteilt:

- Zuerst erfolgte die Verträglichkeitsprüfung, also die Erstellung eines Pflichtenheftes, welches die zu digitalisierenden Prozesse beschrieb indem es untersuchte, welche Prozesse zur Datenverarbeitung bereits vorhanden waren, oder ggf. neu geschaffen müssen.
- Die zweite Phase betraf die organisatorische Vorbereitung und Veränderung, d.h. die Anpassung von Organisationsstrukturen in einer Weise, welche die Verwendung der SOPS überhaupt ermöglicht.
- Darauf folgte die programmtechnische Vorbereitung, mit anschließender Kompilierung der Programme für die jeweiligen Computer-Konfigurationen und die Einführung im Betrieb (Breuning/Hoffmann 1973, 29).

Um die Wechselwirkung verschiedener SOPS untereinander festzustellen, wurden sogenannte Verträglichkeitsuntersuchungen durchgeführt. Beispielsweise benötigte in den frühen 1970er Jahren das SOPS PLUS (Planung und Steuerung der Produktion) häufig als hierarchische Datenbank das SOPS BASTEI (Bankspeicherung technischer Informationen). Die Ingenieure Lothar Breuning und Eckhard Hoffmann schlugen vor, anhand einzelner Produkte die Prozesstauglichkeit zu überprüfen. Das Produkt – ein Boundary Object (Bowker/Star 1999) – könne der Verständigung zwischen den Verantwortlichen für die Einführung der neuen Software und der Belegschaft dienen (Abb. 56). In einem zweiten Schritt führt die Einführung der SOPS jedoch auch zu notwendigen Veränderungen in der Organisation. Die Autoren geben aber auch einen Hinweis darauf, wie die Einführung elektronischer Datenbanken generell mit einer Flexibilisierung der Produktion verwoben ist: »Es ist zweckmäßig, für jede Variante einer Baugruppe bzw. für jedes Erzeugnis eine gesonderte Stückliste zu schreiben. Dadurch können ohne weiteres für jedes Produkt [...] die entsprechenden Stücklistensätze gewonnen und abgespeichert werden« (ebd. 31). Im Umkehrschluss bedeutete dies, dass die elektronische Stücklistenverwaltung einen wichtigen Vorteil ›Neuer‹, beziehungsweise ›kooperativer Medien‹ ausspielen konnte, und zwar Varianz (Manovich 2001, 36–47).

Elektronische Stücklisten für ein bestimmtes Produkt konnten im Vergleich zu vorherigen Verfahren deutlich dynamischer variiert werden, da zu den zusätzlich benötigten Teilen lediglich neue Verknüpfungen in der Datenbank gelegt werden mussten. Varianz wurde also über Beziehungen erzeugt. Die Kosten für konzeptuelle Veränderungen von Produkten sanken damit und die Produktpalette konnte insgesamt variabler eingerichtet werden. Reiner Dollner und Lutz Weickert gaben in *rechentechnik/datenverarbeitung* 1973 ähnliche und weitere Beweggründe an: das generelle Anwachsen der Produktivität, die Überführung von Forschung und Entwicklung in Produktreife in kürzerer Zeit und die Vergrößerung der Betriebe einhergehend mit einer »verstärkte[n] Zentralisierung der Produktion durch Kooperation« (Dollner/Weickert 1973, 11).

Bereits in der ersten von mehreren Umsetzungsstufen des Softwaresystems, so Breuning/Hoffmann, »wird schwerpunktmäßig der Aufbau der Datenbank behandelt, da er sich in starkem Maße nach den Anforderungen der die Datenbank nutzenden Programme, also den systemeigenen Auswertungsprogrammen (Wiederauffindungsprogramme), den nachgelagerten SOPS und evtl. zusätzlichen Anwenderprogrammen, richten muß. Deshalb müssten zu diesem Zeitpunkt die wichtigsten Anforderungen der Nutzer vorliegen. Dabei geht es hauptsächlich darum, welche Dateien aufgebaut werden und welche Dateien sie enthalten sollen« (Breuning/Hoffmann 1973, 30).

Im Anschluss empfahlen die Autoren, frühzeitig eine erste Umsetzungsstufe in Nutzung zu bringen, zum Beispiel die Gegenstandsdatei und Stücklisten. Erst dann sollten in weiteren Ausbaustufen die Module, wie beispielsweise Arbeitsplan- und Arbeitsplatzdatei hinzugefügt werden. Daraus ergebe sich



auch das Verhältnis zu den benötigten Daten: »Weiterhin werden Anforderungen gestellt, für welche Strukturen Daten gespeichert werden sollen und für welche Planzeitraum die Daten welche Qualität und Aktualität besitzen müssen« (ebd. 30).

Das wirft die Frage auf, wie die Daten miteinander zu verknüpfen waren und woher überhaupt Daten gewonnen werden sollten. Die Verknüpfung erfolgte durch den Verweis auf die Ablageorte der Dateien im Speicher und den Ort des Datensatzes innerhalb der jeweiligen Datei. Über Schlüssel, zum Beispiel Produkt- oder Teilnummern, sollte eine Zuordnung erfolgen, wobei überall dort, wo keine existierenden organische Schlüssel vorhanden waren, »systemlose numerische Schlüssel« (ebd. 31) eingeführt werden sollten.

Zur Gewinnung der Daten stellten die Autoren eine Reihe von Kriterien auf, damit ein erhobenes Datum im Sinne des SOPS verarbeitet werden konnte:

- Die Bezeichnung des Datums in der SOPS-Software sollte den betrieblichen Primärbelegen (Papierformularen) entsprechen.
- Die Quelle eines Datums sollte angegeben werden (z.B. betriebliche Primärbelege, oder zentrale, übergeordnete Primärbelege).
- Zum Zweck der Speicherorganisation musste für jedes Datum die Anzahl der Bytes auf dem Festplattenspeicher angegeben werden – ein Umstand der unter relationalen System später überflüssig wurde.

Diese Überlegungen flossen idealerweise in eine prozessorientierte Tabelle, mit der sich der Prozess der Einführung des SOPS verfolgen und steuern ließ.

Tabelle: Angaben zur Gegenstandskartei<sup>1</sup>

Datengruppen	Feldnamen des Namens		Bezeichnung des Datums			Quelle des Datums		Aufbau des Datums							Nutzer des Datums			Bemerkungen					
	auf Ein- gabe- bereich	...	in SOPS Unterlagen	in betrieblichen Primär- belegen	in zentralen Primär- belegen	betriebliche Primärbelege Arbeits- plan	zentrale Primärbelege Arbeits- plan	in betrieblichen Primärbelegen	in zentralen Primärbelegen	auf Eingabe- daten-träger	auf Wechselplatten- speicher	Realisierungsstufe 1	...	SOPS Buster	SOPS PROG	Anwen- derpro- gr. 1	...						
vom Programm- system gebliebene Daten	—	...	Satzzähler oder Positionskette	—	—	—	—	—	—	—	—	—	—	2	P	...	X	X	—	...			
	—	...	Dispositions- stufe	—	—	—	—	—	—	—	—	—	—	2	P	...	X	X	—	...			
	—	...	...	—	—	—	—	—	—	—	—	—	—	...	...	...	X	X	...	...			
auf der Datenbank für das hochgeagerte SOPS bzw. zusätzliche Anwenderprogramme zuzuservierende Felder	—	...	effektiver Bestand	—	—	—	—	—	—	—	—	—	—	5	P	...	—	X	—	...	Standard kann genommen werden		
	—	...	disponibler Bestand	—	—	—	—	—	—	—	—	—	—	5	P	...	—	X	—	...	Standard kann genommen werden		
	—	...	...	—	—	—	—	—	—	—	—	—	—	...	...	...	—	...	...	...	...		
Daten bei deren Fehlen trotzdem unmittelbar oder mittels Parameter des nachgelagerten SOPS bestimmte Algorithmen des nachgelag. SOPS ausgef. werden	GDE VART	...	Vorkaufart der Fertigung	Produktions- verlauf	Verlaufsform	X	...	X	...	1	X	1	X	1	X	1	Z	...	—	X	—	...	pauschale Vorgriffszeit ist zulässig, deshalb über Parameter
	—	...	Vorgriffszeit	—	Storgriffszeit	—	—	—	—	—	—	—	—	—	—	—	—	—	—	X	—	...	...
	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
Daten, die unbe- dingt aus den Primärbelegen entnommen werden müssen	GDE GEGN	...	Gegen- stands- nummer	Sachnummer	Gegenstands- nummer	X	...	X	...	10	xxx xxx-xx	6	xxxxxxx	6	xxxxxxx	6	z	...	X	X	X	...	neue systemlose Gegenstandsnr. notwendig zusätzlich 6. Stelle auffüllen
	—	...	Material- nummer	—	Artikel- nummer	X	...	X	...	5	xxxxxxx	6	xxxxxxx	6	xxxxxxx	6	z	...	X	X	X	...	...
	GDE TKMZ	...	Teilkenn- zeichen	—	in Artken- zeichen	—	—	X	...	—	—	1	X	1	X	1	Z	...	X	X	X	...	neuer Schlüssel notwendig, in PLUS vorgegeben
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

Abb. 56: Gegenstandskartei für SOPS in Tabellenform (Breuning/Hoffmann 1973, 30).

Woher also sollten die Informationen stammen? »Genügen die vorhandene betrieblichen Primärbelege nicht den Anforderungen, ist auf die Einführung der einheitlichen zentralen Primärbelege der

Zentralstelle für Primärdokumentation zu orientieren. Diese sind auf die Anforderungen der SOPS abgestimmt« (ebd. 31). Die Primärbelege stellen eine weitere Schichtung der Materialität der elektronischen Datenbank dar, eine Schichtung, die häufig der Wahrnehmung entgeht, da Daten oft erscheinen, als seien sie anlasslos vorhanden. Aus den bisherigen Ausführungen lässt sich bereits schließen, dass Daten in einem komplexen Prozess der Figuration konstituiert werden und daher ist eine nähere Erkundung des Sediments Primärdokumentation dringend angezeigt.

### 3.2.2.2 Einheitliche Primärdokumentation

Für die USA führt der Historiker Thomas Haigh die Intensität organisatorischer Praktiken mit Tabellen, Formularen und Daten auf die Erfahrungen der militärischen Logistik im zweiten Weltkrieg zurück. »Die ökonomische Mobilisierung während des II. Weltkrieges zog eine unglaubliche Steigerung der industriellen Produktion nach sich und setzte einen Schwerpunkt auf die integrierte Planung von Produktion und Distribution. Pläne zur Arbeitsvereinfachung, gedruckte Formulare, Organisationsdiagramme, Prozessdiagramme und Handbücher wurden in bisher nicht dagewesenem Maße hergestellt. Diese Erfahrung aus Kriegszeiten demonstrierte vielen Managern, was erreicht werden konnte, wenn Organisationsstrukturen und -prozeduren mit dem Ziel spezifischer Ergebnisse aufmerksam gestaltet wurden, anstatt dass sie sich mit der Zeit von allein ergaben« (Haigh 2001, 18).<sup>238</sup> Haigh gibt damit einen Hinweis, auf die Herkünfte dieser managerialen Techniken.<sup>238</sup> Lässt sich diese genealogische Andeutung auch auf die DDR übertragen? In Folge des 2. Weltkrieges fand in der DDR ein Großteil der Forschung bis in die 1980er Jahre hauptsächlich im nicht-militärischen Bereich statt (Sobeslavsky 1996, 56). So hatten sich in den 1960er Jahren im VVB Büromaschinen Erfurt Gottfried Puttrich und W. Rinn mit der Vereinheitlichung von Lochkarten auseinandergesetzt (Puttrich/Rinn 1960).<sup>239</sup> Diese Auseinandersetzung erfolgte mit Hauptbezug auf Erfahrungen in Industrie und Logistik (Deutsche Reichsbahn), wobei zum Teil auch Erfahrungen aus dem 2. Weltkrieg einfließen.<sup>240</sup>

Ausgangspunkt für eine Vereinheitlichung der Primärdokumentation war ein Defizit, welches durch die Staatliche Zentralverwaltung für Statistik der DDR (SZS) im Zuge der Einführung von SOPS konstatiert wurde und zwar, »daß es vielfach nicht möglich ist, die für die Abarbeitung notwendigen Information in der geforderten Qualität und mit einem vertretbaren Arbeitskräftewand zur Verfügung zu stellen« (Dollner/Weickert 1973, 11). Die Autoren Reiner Dollner und Lutz Weickert konstatierten einen Bedarf zur Vereinheitlichung, da in den vorausgegangenen Jahren die Anzahl verschiedenster Primärdokumentationen bei den entsprechenden Druckverlagen »sprunghaft angestiegen« sei (ebd.). Aufgrund dieser Analyse wurde innerhalb der SZS eine eigene Zentralstelle für Primärdokumentation (ZPD) geschaffen, welche die Arbeiten koordinieren sollte. Wie zentral diese Überlegungen waren,

---

<sup>238</sup> Die auf den Krieg bezogene Periodisierung könnte auch aus anderer Perspektive geltend gemacht werden, nicht als Kriegs- sondern als Hochphase der Industrialisierung im frühen 20. Jahrhundert.

<sup>239</sup> Die Artikelserie *Das Betriebsgeschehen in 80 Spalten* erschien in der Zeitschrift *Neue Technik im Büro* in den Jahren 1960/61 in insgesamt 10 Teilen und umfasste Auftragsbearbeitung, Stücklisten, Arbeitsplatzstammkarten, Vorbereitung der Produktion, Grundmittel und Materialabrechnung.

<sup>240</sup> Vgl. *Die erste Informationsexplosion – Die Rolle der Lochkartentechnik bei der Büro-rationalisierung in Deutschland 1910 bis 1939* (Vahrenkamp 2017); In *IBM und der Holocaust* geht Edwin Black der Frage nach, wie die Deportationslisten des Holocaust prozessiert wurden, und welchen Anteil daran die deutsche IBM-Tochter Deutsche Hollerith Maschinen GmbH (Dehomag) mit der Bereitstellung von Maschinen und Personal hatte (Black 2001). Vergleiche auch die Praktiken der Volkszählung in Albert Speer's »Maschinellen Berichtswesen« in *Die restlose Erfassung – Volkszählen, Identifizieren, Aussondern im Nationalsozialismus* (Aly/Roth 1984).

wird dadurch angezeigt, dass sie in das Gesetzesblatt der DDR in eine Reihe von Anweisungen aufgenommen wurden: Zuerst erfolgte die *Anordnung über die Einführung einheitlicher datenverarbeitungsgerechter Primärdokumente des einheitlichen Systems von Rechnungsführung und Statistik* vom 8. Oktober 1968 im Vorlauf der Orientierung auf SOPS und der bereits erwähnte *Beschluß zur Erhöhung der Effektivität und zur Durchsetzung der sozialistischen Rationalisierung bei der Einsatzvorbereitung für die elektronische Datenverarbeitung* von 1971, mit dem die SOPS gesetzlich verankert wurden (Der Leiter der Staatlichen Zentralverwaltung für Statistik 1968; Ministerrat der DDR 1971). 1968 ging es um die »kombinierten Rechnungssätze, Wirtschaftsverträge, und die Belege und Nachweise für die Grundmittelrechnung« (Pfaffenholz 1974, 34). Schon sechs Jahre später gab es eine erneute Neuauflage der Primärdokumente in der *Anordnung Nr. 2 über die Einführung und Anwendung einheitlicher datenverarbeitungsgerechter Primärdokumente*, die 1974 in Kraft trat (Der Leiter der Staatlichen Zentralverwaltung für Statistik 1974). Sie war deutlich weitreichender gefasst, denn sie betraf die »verbindliche Einführung aller Primärdokumente auf dem Gebiet von Rechnungsführung und Statistik sowie der Fertigungsorganisation« und erweiterte den Kreis der Anwender, nämlich »alle wirtschaftsleitenden Organe, Betriebe, Kombinate, Einrichtungen und Genossenschaften« (Pfaffenholz 1974, 34). Aktualisiert wurde diese Anordnung nochmals 1983 mit einer *Anordnung Nr. 3 über die Einführung und Anwendung einheitlicher datenverarbeitungsgerechter Primärdokumente* (Der Leiter der Staatlichen Zentralverwaltung für Statistik 1983).

Bis 1970 entwickelte die Arbeitsgruppe des SZS die einheitlichen Primärbelege unabhängig von Robotron, danach setzte eine Kooperation ein, basierend auf der Erkenntnis, dass die Informationsflüsse über die Arbeitsabläufe zunehmend digitalisiert werden würden. Die Autoren Dollner/Weickert, beide Mitarbeiter in der SZS, stellten den vorangegangenen Prozess als überaus kontrovers dar: »Umfangreich und hart waren die Auseinandersetzungen innerhalb des Entwickler-Kollektivs [der einheitlichen Primärdokumentation] sowie zwischen dem Entwickler-Kollektiv und anderen Partnern. Hinzu kamen die damals erstmals verstärkt auftretenden Impulse durch den Einsatz der EDV« (ebd., 13). Als Fazit der Diskussionen konstatierten sie, dass sich zwar Leitungsstrukturen und Arbeitsprozesse nicht ändern würden, aber die Informationsflüsse – eine Wahrnehmung, der bereits die eingangs zitierten Autoren Breuning/Hoffmann aus Perspektive des SOPS teilweise widersprachen, denn die Informationsflüsse spielten zwangsläufig in die Arbeitsabflüsse hinein. Dollner/Weickert hoben letztlich hervor, dass der reine Austausch der vorhandenen Tabellen mit den neuen Primärbelegen nicht ausreiche, ja »überhaupt keinen Nutzen, sondern nur Aufwand« bringe. Vielmehr wäre ihre Einführung Anlass für die »Einführung effektiver Organisationslösungen der Fertigungsorganisation« (ebd., 13).

Beispielhaft ist hier jener epistemische Zugewinn zu sehen, der durch die Einführung neu organisierter Informationsaufnahme und -darstellung erzielt werden soll. Unter Spannung steht dabei die Verzahnung von Informationsverarbeitung und Prozesssteuerung: Formulare und Tabellen dienten nicht allein dem Eintrag von Ist-Zuständen, wie beispielsweise der Anzahl produzierter Güter in einem bestimmten Zeitraum. Die »Belege mit Weisungscharakter« (ebd. 11) fungierten auch als prozessorganisierende Denkwerkzeuge.

Die Zentralstelle für Primärdokumentation drängte 1974 auf eine maximale Einheitlichkeit der Primärdokumentation: »Die Betriebe sind nicht berechtigt, entgegen diesen einheitlichen Primärdokumenten andere bei der Projektierung von Anwendungen der EDV aufzunehmen. Ausnahmen bilden entsprechende Vereinbarungen einzelner Einrichtungen, die abgestimmt der ZPD [Zentralstelle für Primärdokumentation] vorzulegen sind« (Pfaffenholz 1974, 34). Pfaffenholz berichtet auch, dass zwar die einheitliche Primärdokumentation bereits allgemein angewendet wird, allerdings häufig Probleme

im Datenaustausch verschiedener Struktureinheiten entstanden, da diese die Ausfüllvorschriften nicht ausreichend beachten würden. Angesichts solcher Durchsetzungsprobleme bestimmte die Anordnung Nr. 2 von 1974 außerdem eine Deadline für die Umstellung auf einheitliche Primärdokumentation zum 31. Dezember 1975, verbunden mit der Forderung, »Die Umstellung darf nicht formal durchgeführt werden, sondern sie muß gleichzeitig mit einer Verbesserung der Verwaltungsorganisation verbunden sein« (Pfaffenholz 1974, 34).

Über die Langfristigkeit des Vorhabens waren sich die Akteure bewusst. Die Vereinheitlichung der Primärdokumentation wurde als Prozess auf mindestens 10 Jahre hin angelegt, »da hier geistig formale Arbeiten durch EDV ersetzt werden sollen« (Dollner/Weickert 1973, 13). Im Laufe der Umstellung gab es mehrere Anwenderkonferenzen, auf denen Feedback aus den Betrieben und Rechenzentrum an das ZPD zurück gegeben wurde, d.h. die Vereinheitlichungsbemühung war keine Einweg-Straße sondern die ZPD suchte aktiv nach Anregungen aus der Praxis (Dollner 1981, 1).

Im April 1972 wurden die aus 24 unterschiedlichen Formularen und Tabellen bestehende einheitliche Primärdokumente in das Programm des Vordruckverlags Freiberg aufgenommen (Dollner/Weickert 1973, 15) und 1976 wurde vom ZPD die Vereinheitlichung der Primärdokumente des technologischen Belegsatzes als abgeschlossen erklärt. »Der wesentliche Rationalisierungseffekt der standardisierten Primärdokumente liegt in der Vereinheitlichung von Begriffen, Stelligkeiten, Informationsstrukturen und der äußeren Gestaltung im volkswirtschaftlichen Maßstab« konstatierte Kurt Schildhauer im Nachgang (Schildhauer 1981, 5).

Für die 1980er Jahre wurde der Übergang von den gedruckten Belegen zum Bildschirm-Arbeitsplatz vorhergesagt, bei zunehmend automatisierter, d.h. nicht länger händischer, Erhebung der Daten. »Auf dem Gebiet der produktionsnahen Datenerfassung zeichnen sich Lösungen ab, die die massenhaft benötigten Lohn- und Materialscheine, Auftragsbegleitungskarten und Terminkarten vereinigen bzw. ablösen« (ebd.). Empfohlen wurde an den neuen »Bildschirmarbeitsplätzen«, die gewonnenen Ergebnisse nicht erneut auszudrucken, sondern im »Datenverarbeitungsspeicher« abzulegen und zu verarbeiten. Diese Anmerkung verweist erneut auf die Zentralstellung von Datenbanken in der zukünftig absehbaren Informationsökonomie. »Jede zu erfassende Information sollte nur einmal erfasst und möglichst multivalent verarbeitet werden« (ebd., 7), denn so Schildhauer, jährlich würde durch »Ausfüllen, Ablochen, usw.« in der DDR »Arbeitskräfteeinheiten in Höhe von 10.000 Vollbeschäftigungseinheiten« (ebd., 7) verbraucht werden, die eingespart werden könnten. Auffallend ist, dass Schildhauer zwar die Umsetzung der Belege in Bildschirmarbeitsgeräte empfiehlt, dass aber noch keine Hinweise gegeben werden wie die Papiervorlagen auf die im Vergleich kleineren Bildschirme angepasst werden sollten. Andeuten lässt sich, dass die Bildschirmisierung der Formulare und Tabellen im Rahmen der SOPS unmittelbar auf den Formularen der ZPD aufbaut und daher ähnliche Ästhetiken und Funktionalitäten erzeugt, welche heute die Ästhetik der Datenbank (an ihrer Oberfläche) begründen.

Noch eine weitere Veränderung der Datenerhebungspraxis entnehmen wir dem Artikel: »Mit dem Einsatz von dezentralen Datenerfassungsplätzen ist eine »Nebenbei-Datenerfassung« anzustreben« (ebd.). Damit deutet sich eine zunehmende Dezentralisierung aber auch Verflechtung mit den unmittelbaren Produktionsprozessen an. Zwar ging die Argumentation nach wie vor von Großrechnern und Zugriff per Terminal aus, doch mittelfristig war bereits absehbar, dass an den Arbeitsplätzen nicht allein Bildschirmterminals ohne Rechenkapazitäten stehen würden, sondern die, ab Anfang der 1980er entwickelten, Personalcomputer mit eigenen Central Processing Units.

### 3.2.2.3 Integration in das Einheitliche System Elektronischer Rechentechnik (ESER)

Spezifisch für die Bemühungen um die Sachgebietsorientierten Programmiersysteme in ihrer Zeit ist, dass sie, zumindest der Planung nach, in das ESER-System zur Vereinheitlichung der Rechentechnik des RGW integriert sind. Das Einheitliche System Elektronischer Rechentechnik (ESER) erscheint aus dieser Perspektive nicht allein als ein gemeinsames System der Gerätetechnik. Es fungierte, zumindest in den 1970er Jahren auch als gemeinsames Softwaresystem. So wurden 1974 eine Reihe von SOPS von einer internationalen Kommission als ESER-kompatibel bestätigt. Dazu zählten die SOPS BASTEI, PLUS, MAWI, KOMPASS, AIDOS und OPSI (Jeschke/Redaktion 1974, 1).<sup>241</sup>

ESER beruhte auf einem Beschluss der *Mehrseitigen Regierungskommission für die Zusammenarbeit auf dem Gebiet der Ausarbeitung, Herstellung und Anwendung der Rechentechnik* (MRK), welche 1969 gegründet wurde. Die DDR hatte Mitte der 1960er erkannt, dass sie im Alleingang mit ihren beschränkten Ressourcen in der Entwicklung von Computertechnik auf internationalem Niveau nicht mithalten konnte. In der Folge versuchte sie, ein bilaterales Abkommen mit der UdSSR zu erzielen, vorangetrieben durch den damaligen Staatssekretär für Datenverarbeitung im Ministerrat der DDR, Günther Kleiber. »Er war als Kandidat des Politbüros [der SED] auch in der UdSSR respektiert, und hatte Zugang zu den oberen Stellen. Seinem Wirken ist zu verdanken, dass sich die Beziehungen mit der DDR auf dem Gebiet Rechentechnik gut entwickelten« (Interview Merkel 2018, Fol 7). Statt des gewünschten bilateralen Vertrages, bestand die UdSSR allerdings aus politischen Gründen auf ein multilaterales Abkommen, welches weitere RGW-Länder einbezog und Kern des ESER wurde.

Ausgangspunkt für ESER war zudem eine generelle Unzufriedenheit mit teils widersprüchlichen, unkoordinierten Entwicklungen innerhalb der einzelnen Länder und im RGW. So existierten Anfang der 1960er eine ganze Reihe von Betriebssystemen, deren Programmierung einerseits aufwendig war und andererseits sich jeweils nur auf die vor Ort produzierte Hardware bezog (Interview Merkel 2018, Fol 2). Dies betraf zum Beispiel die Minsk-Computerreihe aus Minsk, die ein anderes Betriebssystem als die ebenfalls in der UdSSR produzierten BESM in Moskau oder die in Pensa produzierten Großrechner der Reihe Ural verwendeten. Der vergleichbare Großrechner Robotron R 300 in der DDR verfügte wiederum über ein anderes, eigenes Betriebssystem. Diese historisch gewachsene Zersplitterung nach Produktionsstandorten setzte sich auch bei kleineren Computern und Rechenmaschinen fort und widersprach der offensichtlichen Ressourcenknappheit in den sozialistischen Ländern. Bis zur Entstehung des ESER und erstaunlich vielen Fällen auch darüber hinaus, kann man davon ausgehen, dass die Technologieentwicklung nicht vertikal von oben nach unten durchgeplant war, sondern dass lokal verankerte Schwerpunkte ihre Interessen in den Planungskommissionen der Länder durchsetzen konnten.<sup>242</sup> Für eine stärkere Zentralisierung der Softwareentwicklung sprach die Erkenntnis, dass die Softwareentwicklung der Betriebssysteme nicht länger der Hardwareentwicklung zu folgen hätten. Stattdessen sollten beide gleichzeitig und integriert stattfinden sollten, um Umwege in der Software, welche

---

<sup>241</sup> Auf der dokumentierenden Website [www.eser-ddr.de](http://www.eser-ddr.de), deren Autor Hans-Georg Jungnickel einer der ESER-Hauptkonstruktoren der DDR war, werden SOPS nicht als Teil von ESER erwähnt, scheinen aus seiner Sicht also eher eine untergeordnete Rolle gespielt zu haben. Auf die gemeinsame Entwicklung von Betriebssystemen geht er jedoch ein, wobei insgesamt gesehen der Hauptfokus der Website auf der Hardwareentwicklung liegt (vgl. auch Jungnickel 1984). Es ist somit davon auszugehen, dass der Hauptanteil der Softwareproduktion im Rahmen von ESER im Bereich der Betriebssysteme lag.

<sup>242</sup> Die DDR hatte mit der Gründung des Kombinars Robotron nahezu alle potentiellen Hersteller unter einem Dach vereint. Ausnahmen blieben vor allem der VEB Elektro-Apparate-Werk Berlin-Treptow, welcher eine eigene Reihe von Prozess-Computern herstellte und der VEB Elektroprojekt und Anlagenbau Berlin, der unter anderem das Prozesssteuersystem Audatec herstellte. Ab Mitte der 1980er kam der VEB Mikroelektronik Mühlhausen hinzu, welcher die Homecomputer der KC-Serie baute.

durch Vorgaben der Hardware erzeugt wurden, zu vermeiden (Interview Merkel 2018, Fol 3). Dies entsprach der neuen Vorgehensweise bei IBM, welche ab den 1960er Jahren die Hardware und Softwareproduktion unter dem Begriff der Systemarchitektur in einer gemeinsamen Arbeitsgruppe zu koordinieren begann – erstmals verwirklicht im IBM OS/360.<sup>243</sup>

In der Mehrseitigen Regierungskommission des ESER fiel die nicht unumstrittene Einigung auf ebendiese Geräte- und Betriebssystemkonfiguration von IBM OS/360 als Vorlage für die eigenen Entwicklungen. Auch dies war Ergebnis einer längeren Aushandlung vor dem Hintergrund der verschiedenen im RGW bereits in Verwendung befindlichen Betriebssysteme.

Schon 1966 diskutierte eine länderübergreifende Arbeitsgruppe des RGW Fragen der Vereinheitlichung. Die VR Ungarn unterbreitete den Vorschlag eines gemeinsamen, zentralen Rechenzentrums, welcher jedoch durch die UdSSR aus Sicherheitsgründen, abgelehnt wurde. Die unterschiedlichen Auffassungen innerhalb der UdSSR spitzen sich schließlich auf einen Konflikt zwischen zwei Konzepten zu: »In der UdSSR herrschten noch unterschiedliche Auffassungen darüber, welches der beiden in der UdSSR favorisierten konzipierten Architekturkonzepte, ›RJAD‹ im Sinne der Empfehlung der RGW-Kommission (vergleichbar mit IBM/360) oder ein System vergleichbar mit ICL 1900 (favorisiert von den URAL-Entwicklern in Pensa), künftig umfassend verfolgt werden sollte. Experten von der Akademie der Wissenschaften der UdSSR (Schura-Bura, Dorodnyzin, Gluschkow) entschieden auch in Abstimmung mit Experten aus der DDR schließlich für RJAD« (Merkel 2005, 44). Die DDR schlug im Rahmen der bilateralen Verhandlungen ebenfalls ein IBM System/360-kompatibles System vor, denn Robotron hatte den Weg in diese Richtung bereits eingeschlagen.

Laut Gerhard Merkel, in diesem Zeitraum Leiter des Instituts für Datenverarbeitung Dresden und ab 1966 abgeordnet als Stellvertreter des Ministers für Elektrotechnik und Elektronik, war den Beteiligten bei Robotron klar, dass den eingeschränkten Personalressourcen für Softwareentwicklung in der DDR nur durch die Nachnutzung vorhandener Software beizukommen sei. »Wir hatten keine Leute für Software und was alles noch dran hängt, [und da] haben wir gesagt, wir halten uns daran, dass es rechen-technisch kompatibel zu IBM ist. Und da war das IBM System/360 schon bekannt« (Interview Merkel 2018, 10).<sup>244</sup> Neben der Nachnutzung der IBM-Software, die oft durch Adaption ›legalisiert‹ wurde, stellten die SOPS eine zweite, größtenteils eigenentwickelte Stütze der Nachnutzung dar. »Dies gibt den einzelnen Partnern die Möglichkeit, ihre Kräfte auf bestimmte Schwerpunkte zu konzentrieren, um damit schneller umfassender Lösungen anbieten zu können und im Austausch dafür andere POS-Bestandteile [Anwendungssoftware – F.H.], die von den Partnerländern entwickelt werden, zu erhalten« (Autorenkollektiv/Gräßler/Freudenberg 1975, 9).

Oberstes Gremium des ESER war der Rat der Chefkonstrukteure, dem eine Reihe von Spezialistenräten unterstanden, besetzt mit Mitgliedern der verschiedenen Teilnahmestaaten. Diese wiederum

---

<sup>243</sup> Die Grundlagen wurden 1961 im IBM-internen Bericht der SPREAD-Task Group, welche die Rechnerentwicklung bis 1970 konzipierte, gelegt (IBM SPREAD Task Group 1961, Abschn VI. A).

<sup>244</sup> Ein neuer Bereich für Softwareentwicklung im VEB Elektronische Rechenmaschinen, Karl-Marx Stadt (heute: Chemnitz) konzipierte ab 1968 das Betriebssystem IBM-kompatible OS/ES für den Großrechner R400. Im Zuge der ESER-Entwicklung wurde R400, der ursprünglich nur DDR-intern als R300-Nachfolger vorgesehen wurde, in ein ESER-Projekt umgewidmet. »Im Unterschied zu zeitgleichen Software-Entwicklungen von Firmen aus Amerika, Fernost und Westeuropa, die ebenfalls an der inzwischen zum Erfolg gewordenen IBM/360-Architektur orientiert waren, standen Software-Quelltexte nicht über Lizenzierung zur Verfügung. Deshalb mussten Basis-Quelltexte durch Re-Assemblierung aus den verfügbaren Maschinenprogrammen gewonnen werden und anschließend Zeile für Zeile untersucht, verstanden und als verwendbarer Quelltext aufbereitet werden. Ohne diesen aufwändigen Arbeitsschritt wäre die Bereitstellung von ESER-Rechnern nicht möglich gewesen« (Körner 2012, Fol 4).

gründeten Arbeitsgruppen mit verschiedenen Themenschwerpunkten, darunter auch Datenverarbeitung und Datenbanken. Dem Rat der Chefkonstruktoren beigelegt war ein ökonomischer Rat und ein Koordinierungszentrum, das als Büro der Mehrseitigen Regierungskommission fungierte und seinen Sitz in Moskau hatte. Jährliche Koordinationstreffen und Schwerpunkt-Ausstellungen von ESER-Technik auf Messen in den Teilnahmeländern dienten der Vernetzung und Kundenpflege.

Die Aufnahme in die ESER-Nomenklatur war gleichbedeutend mit der Exportmöglichkeit zwischen den Teilnehmerländern, weshalb die Zertifizierung der SOPS für Robotron einen wichtigen Schritt darstellte. Als Chefkonstruktoren in den ESER entsandte die DDR Robotron-Mitarbeiter mit Leitungsfunktion, sodass das Kombinat auf hoher Ebene in ESER integriert war.

Die Mitglieder der SOPS-Abnahmekommission Ende März 1974 spiegelten den Stand der ESER-Mitgliedsländer wieder. Vertreten waren neben der DDR, Bulgarien, Ungarn, ČSSR, Rumänien, Polen und die UdSSR. Diese Arbeitsgruppe testete die durch Robotron zur Verfügung gestellten SOPS auf einem ESER ES 1040-Computer, der im Jahr zuvor in die Produktion aufgenommen wurde und zum Kern einer Großrechnerinstallation im VEB Elektromotorenwerk Wernigerode gehörte (Jeschke/Redaktion 1974, 1). Zur Eser-Integration gehörte auch eine Testimplementierung im 1. Moskauer Uhrenwerk S. M. Kirow. In deren Rahmen wurden praktische Probleme überprüft, die sich aus der Übernahme des durch Robotron vertriebenen SOPS BASTEI in die UdSSR ergaben und sich auf Sprache, Dokumentation und die organisatorischen Vorbereitung der Betriebsabläufe für die Softwareverwendung bezogen (Grunert 1975).

Insgesamt waren die SOPS im ESER jedoch ein untergeordnetes Thema. Viele Publikationen zu ESER bezogen sich auf verfügbare Hardwarekonfigurationen.<sup>245</sup> Einige Aufmerksamkeit wurde auch den Betriebssystemen als Standardsoftware gewidmet. Weniger Wahrnehmung hingegen fanden die ASU und SOPS in der Praxis der ESER-Umsetzung, entgegen den Bestrebungen in den Gremien.

### 3.2.3 Dresden, Guben, Moskau, Rostock: Sachgebietsorientierte Programmiersysteme (SOPS)

»Software heißt der Inbegriff solcher Texte, die im Unterschied zu allen bisherigen Schriften der Geschichte das, was sie schreiben, auch tun« (Kittler 1995).

Bis Mitte der 1970er Jahre war in der Fachzeitschrift *rechentechnik/datenverarbeitung* exemplarisch zu beobachten, wie die einheitliche Primärdokumentation in der DDR als Voraussetzung für die Nutzung von SOPS gezielt etabliert wurde. Anschließend veröffentlichten Autoren in den Fachzeitschriften zahlreiche »Berichte aus der Praxis«, welche Fallbeispiele der SOPS-Anwendung dokumentierten. Anhand dieser soll im Folgenden die Einführung von SOPS exemplarisch rekonstruiert werden.

#### 3.2.3.1 Von der Primärdokumentation zur modularen Softwarefiguration

Hoffedank/Grunert vertieften 1973 für die Zentralstelle Primärdokumentation die Thematik in *rechentechnik/datenverarbeitung*: »Eine Stückliste ist die mengenmäßige Zusammenstellung aller Gegenstände

---

<sup>245</sup> Zum ESER beispielsweise siehe (Autorenkollektiv/Münch 1974; Günther 1974; Jungnickel 1984; Jungnickel 2013).

(Materialien, Fertigungsteile, Baugruppen), die erforderlich sind, um ein Produkt (Baugruppe, Erzeugnis) herzustellen. Die Stückliste ist in jedem Fertigungsbetrieb eine der wichtigsten Arbeitsunterlagen. Ihre Bedeutung nimmt mit der Grösse des Produktsortimentes und mit der Kompliziertheit der zu fertigenden Erzeugnisse zu« (Hoffedank/Grünert 1973, 23). Die Autoren arbeiteten in der Folge heraus, warum eine Datenbank für Stücklistenbearbeitung Sinn macht: Bei Zugriff auf die gleiche Datenbasis (die Einzelteile) könnten Stücklisten verschiedene topologische Ordnungen annehmen. Beispielfähig erwähnen die Autoren eine Unterscheidung in funktionelle und strukturelle Topologien. Die funktionelle Topologie sei »in den meisten Fällen der Ausgangspunkt für die Anfertigung der technischen Dokumentation« und die strukturelle Topologie »berücksichtigt den Ablauf des betrieblichen Produktionsprozesses« (ebd.). Die Stückliste war, abstrahiert gesprochen, aufgaben- und prozessorientiert. Sie bildete zudem implizit eine zeitliche Komponente ab.

Weitere Ausdifferenzierungen der Stücklisten erfolgen aufgrund unterschiedlicher Anforderungen aus den Perspektiven der Entwickler neuer Produkte, der Maschineneinrichtung und -optimierung, der Produktionslogistik (Materialbedarf, Arbeitsplanung, Terminkontrolle), allgemeiner Kostenberechnung der Produkte und der Planung von Ersatzteilproduktion und Ersatzteillogistik. Diese Bedürfnisse waren nun zuerst in die Planung der Primärdokumente zu integrieren, damit alle später einmal benötigten Daten erhoben werden konnten.

Die Erstellung der Primärdokumente trug Züge von »Informationsmodellierung«, denn hier wurde ein Modell der betrieblichen Realität entwickelt, welches die einzelnen Vorgänge ihrer potentiellen Berechenbarkeit zuführte: »Bei der Entwicklung des einheitlichen Auftragsbelegsatzes wurde der Informationsbedarf der SOPS mit berücksichtigt und somit eine wesentliche Voraussetzung für die umfassende Einführung der SOPS geschaffen« (ebd., 26).

Ausgehend vom SOPS BASTEI, welches als Datenbank der Stücklisten diente, diskutieren Hoffedank/Grünert die Anbindung weiterer SOPS-Module. Das SOPS PLUS (PLANung Und Steuerung) diente der Produktionsplanung und der Kontrolle und Lenkung der Produktion. Um den Bedarf für ein bestimmtes Produkt zu ermitteln, waren Kennzeichen für Einzelteile (Schlüssel), Artikelklassifikatoren (Kategorie), Mengen und Mengeneinheiten zu erheben. Für den Teilbereich der Lageroptimierung von PLUS wurde jedoch eine deutlich größere Menge an Informationen benötigt, denn die Aufgaben der »Umlaufmittelbindung«, der »Lagerhaltungskosten«, und der »Lieferbereitschaft« beinhalteten zusätzlich zeitliche Komponenten und multiple Abhängigkeiten: Wie lang ist ein Material im Umlauf, bevor es ggf. ins Lager zurück kehrt? Kann ein bestimmtes Teil durch einen Betrieb zu einem Zeitpunkt x geliefert werden, oder nicht? Diese Art von Daten sind als »transaktional« zu kennzeichnen.

Die höhere Komplexität und zeitliche Beziehung spiegelte sich für die »Lageroptimierung« in den aufzunehmenden Daten wieder: Artikelklassifikator (Schlüssel), Lagerort, Kosten pro Gegenstand, Bestand, Lagerzugang, Lagerabgang, Verzugsstückzahl, Sicherheitsbestand, durchschnittliche Schichtdauer\*, Vorlaufzeit für Lagerauftrag, Stückzeit\*, durchschnittliche Normzeit\* und weitere werden abgefragt (ebd., 26). Die mit \* markierten Informationen konnten aus BASTEI übernommen werden, die anderen waren in PLUS abgelegt, eine Trennung die man in relationalen Datenbanksystemen nicht vornehmen würde, dort würden alle Informationen in der Datenbank zentral abgelegt.

Auffällig und neu war die enge Integration der SOPS: PLUS ist ohne BASTEI zwar durchaus zu bedienen, aber der Artikel forderte dazu auf, die Informationsrendite durch Integration abzuschöpfen.



KK-LB	00	1 Lstg-Zeitr. 2	3	4 Gegenstands-Nr.	5	6	7 Menge 8	9 Lstg-Schl. 10 Auftr-Nr. 11 Ko-Fr.	AK		
	K	Datum						Nr. XXXXXXXX			
	01	12 Benennung							13		
	K										
	02										
	03	14 Lstg-Zeitr. 15 AG-Nr. 16 Ko.-	17 Gegenstands-	18 t <sub>s</sub>	19 t <sub>A</sub>	20	21 Menge 22 ME	23			
		Datum Lfd. Nr. / St. Lager AP-Nr.					Lgr.				
	04	24	25	26	27	28	29	30	31		
								32	33		
	05	34 Lstg-Zeitr. Datum	35 AG-Nr. Lfd. Nr.	36 Ko.-St. Menge	37 AP-Pers-Nr. Ausgeber	38	39	40	41	42	43

Abb. 57: Fertigungsbeleg 3.3 – Universalendlosvordruck als Auftragsbegleitkarte mit Erfassung (ab Zeile 5) des Materialverbrauchs und der Arbeitszeit. Ab Zeile 5 kann der Beleg tabellarisch endlos nach unten fortgeführt werden (Büttner/Dollner/Kunitz 1981, 23).

Hoffedank/Grunert motivierten erneut zur Reorganisation der Fertigungslenkung und Informationsaufnahme mit dem Ziel der Rationalisierung: »Fällt die Einführung der neuen Belege zeitlich mit dem Übergang zu neuen Organisationsformen zusammen, so können die vorhandenen maschinenlesbaren Informationen in die Belege der Endlosvordrucke übertragen und somit der Erfassungsaufwand minimiert werden« (ebd., 27). Um zu verdeutlichen, dass es sich nicht um eine vereinzelte Praxis handelte, soll exemplarisch der 1981 veröffentlichte tabellarische Endlos-Fertigungsbeleg 3.3 besprochen werden (Abb. 57). Er konnte mit den Belegen 3.1 oder 3.2 kombiniert werden, welche als Auftragsbelegkarten, Stücklisten, Terminkarten, Lohnschein, Materialschein oder Arbeitsplatzstammkarte konzipiert waren.<sup>246</sup> Im Fertigungsbeleg 3.3 waren die Kopfzeilen (1–4) großzügig ausgelegt, um unterschiedliche Verwendungen zu ermöglichen. Sie dienten einer Einordnung der Funktionalität des Belegs. Die nun folgenden Spalten erfassten die Auftragsleistung, und verzeichneten Datum, Arbeitsgang-Nummer, Menge, Personalnummer, Zeitaufwand – transaktional angelegte Daten. Ein Teil der Spalten wurden vorab per Computer mit den vorgesehenen Arbeitsschritten (jeweils eine Zeile pro Arbeitsschritt) bedruckt, die mit einer Stücklistensoftware generiert worden waren.

Nach Auftragsbeginn trennten die Verantwortlichen die Terminkarte für die »Vordisposition«, also die Zusammenstellung der benötigten Teile, ab. »Anschließend gelangt der Endlosvordrucksatz [das heißt alle zum Auftrag zugehörigen zusammengefassten Einzelscheine ab Zeile 5 – F.H.] zu der Kostenstelle des ersten Arbeitsganges, in der durch die Werk tätigen der Materialschein entnommen und das Ausgangsmaterial aus dem Lager geholt wird. Nach der Beendigung des ersten Arbeitsganges trennt der Werk tätige den Lohnschein ab und gibt ihn zur Verrechnung. Die Arbeitsbegleitkarte wechselt mit den verbleibenden Lohnscheinen und den angearbeiteten Teilen zur Kostenstelle des nächsten Arbeitsganges« (Büttner/Dollner/Kunitz 1981, 25). Dies wurde sooft wiederholt, bis alle zum jeweiligen Auftrag (und damit verbunden dem Satz an Fertigungsbelegen) zugehörigen Auftragsbegleitkarten aufgebraucht waren. Die letzte Karte enthielt die Fertig-Meldung.

Die Tabelle diente hier als Prozesswerkzeug und als Kommunikationsmittel zur Steuerung von Arbeitsabläufen. Sie war Ausgabemedium elektronischer Datenverarbeitung und nahm gleichzeitig neue

<sup>246</sup> Standardisiert waren die Fertigungsbelege in der TGL-Norm 31490/01-03 vom 31.3.1976 durch das Amt für Standardisierung, Meßwesen und Warenprüfung, dem DDR-Pendant des DIN-Institutes.

Daten auf, im konkreten Falle beispielsweise durch Fertig-Meldungen der einzelnen Arbeitsschritte, die wiederum in die elektronische Datenverarbeitung eingespeist wurden.

Die beschriebenen Vorgehensweisen werfen Fragen hinsichtlich ihrer Rigidität auf: Geschildert werden in den Artikeln Vorhaben, Anweisungen und Zielvorgaben. Wurden die Vorgaben als sinnvoll empfunden oder als einengend? Wurden sie von den Direktoren und Fachleitern unterstützt oder ignoriert, beispielsweise weil sie Wissen transparent machen konnten, dessen Zurückhaltung die eigene Machtbasis stärkte? Waren die zuständigen Abteilungen ausreichend geschult und motiviert, die Vorgaben in den Fachabteilungen gegebenenfalls auch gegen Widerstand oder Desinteresse durchzusetzen? Wurden die neuen Vorgaben vermittelt, indem Druck ausgeübt wurde, oder indem die dahinterstehenden Motivationen den Beteiligten sinngebend erklärt werden konnten? In welchem Maße die mit der Primärdokumentation einhergehende Formalisierung von Informationen zum Betriebsablauf in der DDR-Realität tatsächlich produktiv wurden, hing von den einzelnen Betrieben, deren Leitung und Belegschaften ab. Das soll im nächsten Abschnitt befragt werden.

### 3.2.3.2 Konzeption der SOPS

»Was sind SOPS, Typenprojekte und Software?« fragte 1974 in einem Artikel in *rechentechnik/datenverarbeitung* rhetorisch Bernd Schmitt aus Rittersgrün. Die Frage beantwortete Prof. Dr. Kurt Sack von der Hochschule für Ökonomie, Berlin: »In den hochentwickelten kapitalistischen Industrieländern werden vor allem von Herstellerfirmen von EDVA eine Reihe von Arbeiten geleistet, die den Forderungen der Kunden nach einer Unterstützung bei der Vorbereitung des Einsatzes und dem Betrieb einer EDVA Rechnung tragen sollen. Die in diesem Zusammenhang erbrachten Leistungen werden allgemein als Software bezeichnet. Diese Leistungen sind jeweils auf die Erzeugnisse eines betreffenden Herstellers bezogen, so daß eine Vielzahl von Doppelentwicklungen sowohl bei der sogenannten Systemsoftware als auch der anwendungsorientierten Software zu verzeichnen sind« (Sack 1974, 28). Dem gegenüber positionierte der Autor die SOPS, als »variierbare Prinzipienlösungen« (ebd.). Einmal erstellte Anwendungen für Lohnbuchhaltung oder Lagerhaltung sollten in verschiedenen Kontexten modular einsetzbar und den Bedürfnissen der jeweiligen Betriebe und Institutionen anpassbar sein. Sacks Beschreibung entsprach durchaus der Realität, denn jenseits der monopolartigen Dominanz von IBM und deren Betriebssystemen gab es in den kapitalistischen Staaten zahlreiche Hersteller, deren Vorgehen untereinander nicht, oder nur äußerst lose koordiniert war, sodass Software tatsächlich unkoordiniert und mehrfach produziert wurde, abhängig vom Marktgeschehen.<sup>247</sup>

In der DDR-Terminologie der 1960er–1980er Jahr wurde Software verschieden klassifiziert:

- Maschinenorientierte Systemunterlagen (MOS) war Software für das Betriebssystem
- Verfahrenorientierte Programmierpakete (VOPP) dienten der teils kybernetisch orientierten Operational Research, also der Optimierung von Verfahren und Logistiken mittels spezifischer Algorithmen
- Problemorientierte Systemunterlagen (POS) war allgemeine Anwendungssoftware.
- Sachgebietsorientierte Programmiersysteme (SOPS) waren POS, die sich auf spezifische Sachgebiete bezogen.

---

<sup>247</sup> Standardisierungsbemühungen traten in den westlichen Staaten häufiger auf, als es sich Sacks Argument entnehmen lässt, so zum Beispiel die Standardisierungsvorhaben der Programmiersprachen ALGOL-60 ab 1958, und COBOL ab 1959, sowie in Bezug auf Datenbanken DBTG/CODASYL ab 1965, und die ANSI-SPARC-Architektur ab 1975. Im RGW treten Vereinheitlichungstendenzen erst durch das gemeinsame ESER-Programm ab 1969 zu Tage, Sacks Diskussion ist also stärker Abgrenzungsbemühung als die Wiedergabe von Realität.

Ab Ende der 1960er Jahre propagierten die Wirtschaftsführung im Zentralkomitee der SED und die Ministerien angesichts der wirtschaftlichen Realitäten die Zielvorgabe, dass Hardware und Software stärker zu vereinheitlichen seien, um Synergie- und Netzwerkeffekte zu erzeugen. So sollte in den Betrieben weniger Software eigenentwickelt werden, sondern die zu den Geräten ausgelieferte Software Verwendung finden.<sup>248</sup> Für Aufgaben wie Lohnbuchhaltung, Stücklisten, Materialwirtschaft war diese mitgelieferte Software jedoch nur beschränkt ausreichend. Größtenteils entsprach der ausgelieferte Softwareumfang Hilfsprogrammen zur Dateiverwaltung, Sortierung, Speicherung auf verschiedene Datenträger und ähnlichem. Für einen domänenübergreifender Einsatz war diese Ausstattung ungeeignet. Daher kam der Vorschlag modularer, variabler Softwarepakete, welcher durch den Robotron-Fachbereichsleiter Rolf Gräßler am 7. Januar 1972, an der TU Dresden, im Zuge einer Dissertation publiziert wurde, zur richtigen Zeit.

Die Sachgebietsorientierten Programmiersysteme (SOPS) wurden innerhalb von Robotron durch Arbeitsgruppen konzipiert und diskutiert. Dann stimmte sie die mehrseitige Regierungskommission (MRK) für einheitliche Rechentechnik im Zuge von ESER untereinander ab (vgl. Abb. 58). Die Dissertation ist somit nicht als eine Einzelleistung eines akademischen Forschers einzuordnen, sondern als begleitende Publikation der Konzeptualisierung der Sachgebietsorientierten Programmiersysteme (SOPS).

Rolf Gräßler war 1963 als Vertreter des Ministeriums für Hoch- und Fachschulwesen in die Regierungskommission für die Grundlagen der Datenverarbeitung in der DDR berufen worden und begann gleichzeitig seine Berufstätigkeit beim Zentralinstitut für Automatisierung (ZIA). Die Beratungen der Regierungskommission flossen in den Beschluss des Ministerrats zur Datenverarbeitung von 1964 ein,<sup>249</sup> welcher im ZIA als Begründung diente, sich mit Problemorientierten Systemunterlagen (Anwendungssoftware) auseinander zu setzen. Zwei Konzeptionen wurden grundsätzlich diskutiert: Der erste Vorschlag betraf die Weiterentwicklung des in der Praxis mit mechanischen Buchungsmaschinen und Lochkarten entwickelten Datenverarbeitungsprozesses *Betriebsgeschehen in 80 Spalten* (Puttrich/Rinn 1960; vgl. auch Bode 1968). Dieses am VEB Nahrungs- und Genussmittel-Maschinenbau (Nagama) entwickelte Verfahren erlaubte die Auftragsverwaltung und Buchung von Geschäftsvorfällen mit durch Schlüssel einander zugeordneten Lochkarten. Die in der DDR verwendeten Lochkarten entstammten der Technologie der Deutsche Hollerith Maschinen Gesellschaft, und waren daher IBM-kompatibel. Der Vorschlag bestand darin, das bewährte, bis dahin elektromechanische, Vorgehen in die elektronische Datenverarbeitung zu überführen. Als zweites diskutierte man die Schaffung von Typenlösungen, welche sich von Beginn an stärker an der elektronischen Datenverarbeitung orientieren und auf bestimmte Typen von Betrieben zugeschnitten sind. Die Entscheidung fiel für diesen neueren Ansatz. Jedoch verlief die Programmierung der Typenlösungen alles andere als reibungslos: Es entstanden Probleme aufgrund fehlender Abstimmung zwischen der organisatorischen Leitung und den Programmierern, was zu Verzögerungen an der Grenze zum Scheitern führte. Mitte 1970 war die Programmierung der Typenlösungen beendet, und sie wurden beispielsweise in Betrieben des Maschinenbaus, der Bauindustrie und der Chemie eingesetzt. Rolf Gräßler resümierte drei Jahrzehnte später: »Bald mußten wir demnach einsehen, daß die Entwicklung von Typenprojekten aus zwei Gründen in eine Sackgasse führen würde.

---

<sup>248</sup> Flankiert war dies seit 1971 durch einen *Beschluß zur Erhöhung der Effektivität und zur Durchsetzung der sozialistischen Rationalisierung bei der Einsatzvorbereitung für die elektronische Datenverarbeitung*, demnach Eigenentwicklungen eingeschränkt werden sollen, und Betriebe auf vereinheitlichte Software zurückgreifen sollen (Ministerrat der DDR 1971).

<sup>249</sup> Vgl. *Programm von Maßnahmen zur Entwicklung, Einführung und Durchsetzung der maschinellen Datenverarbeitung in der DDR in den Jahren 1964 bis 1970* (Ministerrat der DDR/Politbüro des ZK der SED 1964).

### Sachgebietsorientierte Programmiersysteme (SOPS) / Datenbanken

Bankspeicherung technischer Informationen	BASTEI (Adaption von IBM BOMP)	Robotron
System zur automatischen Wiederauffindung von Informationen	SAWI	Robotron
Datenbanksystem Robotron	DBS/R	Robotron
Automatisiertes Informations- und Dokumentationssystem	AIDOS	Robotron
<b>Sachgebietsorientierte Programmiersysteme (SOPS)</b>		
Materialwirtschaft	MAWI	Robotron
Planung der technischen Vorbereitung der Produktion	TEWO	ESER-Partner
Planung und Steuerung der Produktion (operative Produktionsplanung) mittlere Serienfertigung	PLUS	Robotron Kooperation mit ESER-Partner
Komplexe Betriebsplanung (technisch-ökonomische Planung)	KOMPASS	Robotron
Grundmittelwirtschaft (Inventar)	GRUMI	Robotron
Investitionsrechnung	INVEST	
Absatzlenkung	ABSATZ	VEB Robotron-Vertrieb Berlin/Leipzig
Planung und Abrechnung der Arbeitskräfte	PAAK	VEB Robotron-Vertrieb Berlin/Leipzig
Kostenrechnung	KORAST	
Datenbank für Kosten- und Leistungsrechnung	KOLDA (mit DBS/R)	Robotron
Kontokorrentrechnung	KOKO	Robotron
<b>Verfahrensorientierte Programmierpakete (VOPP)</b>		
Optimierung mittels Simplextechnik	OPSI	
Diskrete Optimierung, Transportoptimierung	Diskrete Optimierung	
Mathematische Statistik	Statistik	
Simulation von Systemen mit diskreten Ereignispunkten	SIMDIS	
Netzplantechnik	Netzplantechnik	
Numerische Mathematik	Numerische Mathematik	

Abb. 58: Übersicht über die sachgebietsorientierten Programmiersysteme (SOPS), Stand 1975. Für verschiedene Rechnergenerationen standen unterschiedliche Entwicklungen und Weiterentwicklungen zur Verfügung. Die Verfahrensorientierten Programmierpakete (VOPP) waren stärker auf Algorithmen im Sinne der Operations Research als auf Datenverarbeitung ausgerichtet und verfügten über vereinheitlichte Daten-Schnittstellen, untereinander und zu den SOPS (Autorenkollektiv/Gräßler/Freudenberg 1975).<sup>250</sup>

<sup>250</sup> Diese Auflistung zeigt eindrucksvoll das Verhältnis bürokratisch-kooperativer und kybernetischer Praktiken. Während genealogisch gesprochen die Sachgebietsorientierten Programmiersysteme (SOPS) den kooperativen, transaktionalen und Informationsorientierten Praktiken entstammen, knüpften die Verfahrensorientierten Programmierpakete (VOPP) stärker an die kybernetischen Diskurse von Kontrolle, Algorithmus, Optimierung, Netzwerk und Signalverarbeitung an. Für die VOPP, die nicht im Zentrum der Untersuchung standen, da sie nicht zum Aufgabengebiet von Robotron gehörten, verweist Gräßlers Dissertation auf ein nicht näher bezeichnetes Projekt zur »sozialistischen Operationsforschung«, möglicherweise sind damit die Bestrebungen an der HU Berlin um Klaus Fuchs-Kittowski gemeint (vgl. Fuchs-Kittowski/Tschirschwitz/Wenzlaff 1973; Fuchs-Kittowski u. a. 1976). In der Sowjetunion war die der Kybernetik entstammende Operationsforschung

Einerseits schien die Trennung von Organisatoren und Programmierern für die Ausarbeitung von Datenverarbeitungsprojekten nicht geeignet, andererseits ließen sich die für einen bestimmten Betrieb ausgearbeiteten Typenprojekte in ihrer starren Form nicht auf einen anderen Betrieb übertragen« (Gräßler 2006, 290). Scheiterte der Vereinheitlichungs- und Rationalisierungswunsch?

Wenn sowohl das bewährte Lochkartensystem als auch die neuen Typenprojekte den notwendigen Grad an Komplexität nicht abbilden konnten, welches Vorgehen würde dann weiterführen? Die Auswertung der problematischen Erfahrungen führte zu einem neuen, innovativen Ansatz: »Die allgemeingültige Software muß variabel gestaltet werden, sie muß eine Problemvariabilität gewährleisten und auf eine gemeinsame Datenbank zurückgreifen« (ebd.). Hier wird eine signifikante Schwelle überschritten, in der Erkenntnis, dass die, mit der Problemvariabilität einhergehende, Variabilität der Software den Bedarf an elektronischen Datenbanken als Werkzeug der automatisierten Informationsverarbeitung begründet. Auffallend ist, dass die Datenbank als gemeinsame Ressource eine konzeptuelle Zentralstellung erfährt. Diese Zentralstellung führt jedoch nicht dazu, dass die elektronische Datenbank in den Vordergrund der Aufmerksamkeit rückt – vielmehr weicht sie in den infrastrukturellen Hintergrund. Sie wird Teil eines komplexeren Systems, eingebettete und ›unsichtbare‹ Voraussetzung, welche das Funktionieren der Gesamtfiguration garantiert (vgl. Star/Ruhleder 1996, 5f.).

Die SOPS-Dissertation wurde 1972 in der Sektion der Sozialistischen Betriebswirtschaft der TU Dresden abgelegt. Einleitend argumentierte die Dissertation, dass »die Produktivkräfte die Kraft und Fähigkeit der Gesellschaft verkörpern, die notwendigen Mittel zur Befriedigung ihrer Bedürfnisse mit einem relativ geringen Aufwand an gesellschaftlicher Arbeit zu produzieren« (Gräßler 1972, 2). Der möglichst geringe Aufwand zur Bedürfnisbefriedigung solle durch staatliche zentrale Planung erreicht werden, so der Autor mit Bezug auf Walter Ulbrichts Schlussansprache auf dem VII. Parteitag der SED mit dem Titel *Die gesellschaftliche Entwicklung in der DDR bis zur Vollendung des Sozialismus*.<sup>251</sup> Dabei müsse die Datenverarbeitung auf einem »wissenschaftlich begründeten Leitungs-, Informations- und Modellsystem« basieren. In der Dissertation sollte dieses »wissenschaftliche« Programm formuliert werden. Als Hauptproblem identifizierte die Studie, dass zwar die Geräte durch die Hersteller zur Verfügung gestellt würden, aber die benötigte Software durch die Anwender selbst programmiert werden müsste. Dabei seien zwei verschiedene Vorgehensweisen zu erkennen:

---

u.a. von Anatolij I. Kitov vorangetrieben und beschrieben worden, zum Beispiel in seinem Buch *Программирование экономических и управленческих задач* (dt. in etwa: Die Programmierung ökonomischer und informationeller Aufgaben). Es erschien 1971 und schilderte Kitovs Erfahrungen mit der Programmierung des Leitungsinformationssystems (ASU) für das Ministerium für Funkindustrie, welches für weitere Ministerien der Verteidigungsindustrie als Vorbild diente (Kitov 1971). Gräßlers Dissertation erwähnte Kitov nicht, verwies jedoch auf Georg Klaus' Buch *Kybernetik und Gesellschaft* (1964). Über die Kybernetikdebatte der 1960er Jahre in der DDR berichtete einleitend der Historiker Segal, der sie in einen Zusammenhang mit der Neuen Ökonomie des Sozialismus (NÖS) stellt, die als Wirtschaftsreform in der ersten Hälfte der 1960er Jahre diskutiert wurde (Segal 2001, 61ff.). Beide, NÖS und Kybernetik, wurden spätestens mit der Machtübergabe an Erich Honecker 1971 beendet. Das Ende der Kybernetik-Debatte öffnete allerdings den Raum für die pragmatisch orientierte »Elektronische Datenverarbeitung« (ebd., 64). Klaus Fuchs-Kittowski schilderte dies 2007 als *Herausbildung von Sichtweisen der Informatik in der DDR unter Einfluss der Kybernetik I. und II. Ordnung* (Fuchs-Kittowski 2007). Eine generelle Einführung zur widersprüchlichen Rezeption der Kybernetik in der UdSSR legten (Gerovitch 2002) und (Peters 2017, 15–80) vor. Die Operationsforschung in der ČSSR stellt (Manas 1995) dar.

<sup>251</sup> Die Dissertation erwähnt es nicht, und setzt sozusagen stillschweigend die Kenntnis der Leser voraus, dass im Kapitalismus die Bedürfnisbefriedigung im Zuge der maximalen Ausbeutung individueller Arbeitskraft und des gegenseitigen Unterbietens der Einzelkapitalien erfolge. Erst vor dieser Folie wird die Aussage markanter.

- Eine genau auf die jeweiligen Bedürfnisse angepasste ›Typenlösung‹, die für genau ein Informationsbedürfnis an einer Lokalität zu Verfügung stehe. Wollte man sie anderweitig weiterverwenden, müssten Produktionsabläufe der vorhandenen Typenlösung umfangreich angepasst werden. Für Typenlösungen gäbe es durchaus berechnete Anwendungen, zum Beispiel bei den staatlichen Expeditionen. Diese waren in den 1960er Jahren nach geographischen Bezirken der DDR unter der Bezeichnung Bezirksdirektionen für Kraftfahrverkehr organisiert. Eine einmal programmierte Typenlösung konnte in mehreren Bezirksdirektionen eingesetzt werden, da hier ein großer organisatorischer Wiederholungsfaktor als Voraussetzung bestanden habe. Typenlösungen bildeten also einen spezifischen Problemtyp ab und konnten überall dort nachgenutzt werden, wo sich dieser Problemtyp wiederholte. Diese Wiederholbarkeit führte zu Einsparungen, da die Software mehrfach wiederverwendet wurde. Dem standen die Kosten gegenüber, welche gegebenenfalls notwendige Anpassungen der Organisationsstruktur an die Software hervorriefen.
- Der zweite Ansatz sei die ›Problemlösung‹, welche Teillösungen modular kombinierbar zur Verfügung stelle und gut dokumentiert eine Verwendung erlaube, in der Produktionsabläufe minimal re-strukturiert werden müssten (ebd., 7). Dies sei eine Vorgehensweise, die den Bedürfnissen der Industrie angepasst sei, deren Einsparungspotenzial aber schwieriger zu schätzen sei, als bei Typenlösungen.

Mit diesen grundsätzlichen Unterscheidungen in ›Typenlösungen‹ und ›Problemlösungen‹ stellt sich die Frage, wie ein derart massives Investment in Software gerechtfertigt werden könne. Wie könnte Gräßler gegenüber seinen Vorgesetzten und den Planbehörden die zahlreichen Arbeitsstunden rechtfertigen, welche die Robotron-Programmierer für die SOPS einsetzen würden?

Die Dissertation entfaltet eine Einschätzung der Einspar-Effekte in der Softwareprogrammierung von problem-orientierter Software, um diese zu Typenprojekten vergleichbar zu machen. In dieser Analyse ist der Einsatz von Software eine Frage des Maßstabes: Einmal entwickelte Lösungen sollten skalierbar werden, um in verschiedenen Kontexten anwendbar zu sein. Um diese Skalierbarkeit zu erreichen, war ein hoher Grad an Standardisierung nötig. Im Fall der Typenlösungen bedeutete dies die Standardisierung der organisatorischen Abläufe in den Betrieben. Für die SOPS hingegen bedeutete es eine Standardisierung der Software und der Schnittstellen zwischen einzelnen Softwaremodulen zu bestimmen, die den Betriebsabläufen anpassbar waren.

Anhand einer Reihe von DDR-Betrieben untersuchte die Dissertation die häufigsten bereits vorhandenen Problemlösungen und deren Einsparpotenziale und unterschied dabei nach einer Reihe von Sachgebieten: Technische Planung, Produktion, Arbeitskräfte, Material, Grundmittel, Kosten-Leistungen-Finanzien und Absatz. Den größten Anteil an Rechnerzeit verortete er in den Bereichen Planung und Steuerung der Produktion (40,7%) und Materialwirtschaft (24,4%) (ebd., 50), und damit in klassischen Material Requirements Planning (MRP)-Anwendungen des Jahrzehnts.

Materialplanung und -abrechnung wurde beispielsweise in der PKW Produktion bei VEB Sachsenring Zwickau, im VVB Werkzeugmaschinenbau Karl-Marx-Stadt (heute Chemnitz), im Büromaschinenwerk VEB Optima Erfurt (später VEB Robotron), im Moskauer metallverarbeitenden Betrieb Freser, im VEB Eisenhüttenkombinat Ost und weiteren durchgeführt (ebd. 44, Tabelle 11). In all diesen Firmen war bereits entsprechende Typensoftware auf einem Großrechner Robotron R 300 im Einsatz. Für die Entwicklung, Installation und Inbetriebnahme der vorhandenen Lösungen ermittelte Gräßler im Durchschnitt eine Dauer von 4,25 Jahren vom Aufstellen der Geräte bis zur tatsächlichen produktiven Nutzung (ebd. 29).

Welche weiteren Argumente zugunsten einer ›Problemlösung‹ konnten in der Dissertation gemacht werden? Ein Vergleich mit westlichen Produkten würde sicherstellen, dass das geplante Investment mit dem ›internationalen Niveau‹ vergleichbar wäre. Der mittlere Teil der Untersuchung besteht daher aus der Evaluation des IBM Lagerverwaltungs-Programms Modular Inventory Control System (MINCOS) und dem Vergleich mit dem SOPS-Konzept, denn: »Nach dem Einsatz von IBM-Anlagen in der DDR wurde teilweise die Meinung vertreten, bestimmte von IBM angebotene Modularprogramme auf dem Gebiet der Planung und Leitung der Industrie unverändert zur Lösung von Aufgaben für die sozialistischen Betriebe der DDR einzusetzen« (ebd., 78). Der Autor argumentierte dagegen. Neben der Verwendung verschiedener Standards, zum Beispiel zehnstelliger Artikelnummern in MINCOS versus, einer fünfzehnstelligen Artikelnummer in der DDR, zählte die Studie zu den Problemen die Abwesenheit einer Materialdisposition (d.h. -vorausplanung) und insgesamt einen großen Umfang notwendiger Anpassungen, um das konzipierte Leistungsniveau von SOPS zu erreichen (ebd., 89).<sup>252</sup> Ein wichtiger argumentativer Aspekt wurde damals anscheinend übersehen: Das Lagerverwaltungssystem MINCOS verfügte über keine standardisierte Schnittstelle zum Datenaustausch mit anderen IBM Programmen, zum Beispiel mit dem IBM Modular System for Computation of Requirements (MOSCOR), einer Stücklistenauflösung, die als Bedarfsrechnung fungierte. In den SOPS hingegen ist diese Schnittstelle mithilfe der Datenbanken BASTEI und SAWI, sowie später DBS/R, im Prinzip angelegt. Allerdings benötigte es bis Anfang der 1980er Jahre, dass sich diese Erkenntnis innerhalb von Robotron und unter den DDR-Kunden tatsächlich durchsetzte.

Zusammenfassend schloss die Studie, dass SOPS grundsätzlich geeignet wären, um Probleme der Planung und Steuerung zu bewältigen, dass ›kapitalistische‹ Software keinen genügenden Funktionsumfang zur Verfügung stelle, dass SOPS einen einmaligen großen Aufwand in der Programmierung benötigen und empfiehlt für die weiteren Abschätzungen die Beispielrechnung anhand des SOPS BASTEI (ebd. 121).

Um die erhofften Effekte zu erreichen, identifizierte die Untersuchung eine Reihe von Zielsetzungen. SOPS sollten unter anderem

- für Branchenfachleute mit Computergrundkenntnissen bedienbar sein,
- modular und erweiterbar angelegt sein,
- auf einer gemeinsamen Datenbasis, also einer Datenbank, aufbauen,
- und mehrsprachig angelegt sein, um im ESER-Rahmen eingesetzt werden zu können (ebd. 126–130).

Die Ausarbeitung von SOPS führte der Autor der Dissertation zurück auf einen Beschluss, der am 7. Mai 1970 im Ministerrat der DDR unter dem Titel *Grundrichtung für die Anwendung von elektronische Datenverarbeitungsanlagen und Prozeßrechnern zur komplexen sozialistische Automatisierung und Rationalisierung im Perspektivplanzeitraum 1971–1975* gefasst wurde (ebd., 140).<sup>253</sup> Ein Blick in diesen Beschluss bezeugt, dass die Hauptrichtung die Automatisierung der Produktion durch digital verknüpfte Fertigungstechnik war, welche »moderne Fließverfahrenszüge mit einer geringen Anzahl an

---

<sup>252</sup> Im Rückblick erscheint dieser Vergleich stark zum Vorteil des SOPS-Konzeptes gewählt, da MINCOS nicht annähernd für den Funktionsumfang konzipiert scheint, wie es die SOPS erreichen sollten. Das deutlich umfassendere Modularprogramm IBM COPICS, stand erst im Laufe des Jahres 1972 im Westen als Dokumentation zur Verfügung. Die Verteidigung der Dissertation erfolgte am 7. Januar 1972, daher ist davon auszugehen, dass COPICS für die Dissertation noch nicht relevant war.

<sup>253</sup> Vgl. *Beschluß über die Grundrichtung für die Anwendung von elektronischen Datenverarbeitungsanlagen und Prozeßrechnern zur komplexen sozialistischen Automatisierung und Rationalisierung im Perspektivplanzeitraum 1971-1975* (Ministerrat der DDR 1970).

Prozeßstufen« (Ministerrat der DDR 1970, 2–4) ermöglichen sollte. Gefordert wurde eine »permanente prognostische Arbeit« (ebd.) an Automatisierungskonzepten, die in Anwendungen zu überführen seien. Ein Schwerpunkt müsse auf »wissenschaftlich begründete Leitungs-, Informations- und Modellsysteme« gelegt werden (ebd.). Diesem Beschluss folgte 1972 eine konkretisierende Direktive des ZK der SED zum Fünfjahrplan 1971–1975.<sup>254</sup> Unter Bezugnahme auf diese Direktive verdeutlicht die Dissertation die modularen Anwendungsgebiete der SOPS: Die »materiell-technischen Versorgung« werde umfassend durch das SOPS MAWI abgedeckt. Eine »Verbesserung der Organisation und Produktion« werde durch das SOPS PLUS erreicht. Das SOPS BASTEI diene der »Rationalisierung der Produktionsvorbereitung«. Die SOPS PLUS, MAWI, PAAK, GRUMI, KORAST und ABSATZ kamen zum Einsatz für die »Kontrolle und Abrechnung der Plandurchführung« und die SOPS KOMPASS und PLUS für die »Beherrschung entscheidender Verflechtungs- und Kooperationsbeziehungen« in den VEB (Gräßler 1972, 144f.).

Abschließend prognostizierte die Dissertation die Anwendungshäufigkeit und Nutzeffekt und ermittelte einen Betrag von 4,2 Milliarden DDR-Mark, die durch Nachnutzung allein in Industrie und Bauwesen eingespart werden können.<sup>255</sup> Diese Zahl bezog sich allein auf Einsparungen in der Erstellung und Anpassung von Softwarelösungen, denn die mögliche Steigerung der Arbeitsproduktivität in Folge der Digitalisierung war aus methodischen Gründen nicht eingerechnet (ebd. 154f.).<sup>256</sup>

In der Einheitlichkeit der Ansätze sah der Autor den Vorteil der sozialistischen gegenüber der kapitalistischen Produktionsweise und konstatierte: »Die Basis für die Ausarbeitung der SOPS in der DDR ist das ökonomische System des Sozialismus, verbunden mit den volkswirtschaftlichen Informationssystemen – einheitliches Planungssystem – einheitliches System von Rechnungsführung und Statistik – Informationssystem Wissenschaft und Technik« (ebd., 159). Dieses Fundament der Vereinheitlichung würde einen höheren Grad der Wiederholung nach sich ziehen. Im Vergleich zum IBM-System PICS (Production Information and Control System) und General Electric-Bulls LAMDA könne »bei planmäßigem Verlauf der konzipierten Arbeiten zum wissenschaftlichen Höchststand auf dem Gebiet der Programmiersysteme für Planung und Leitung aufgeschlossen« werden, was einer Mitbestimmung des Weltstandes entspreche (ebd., 160ff.)

Die in der Dissertation erwähnten Typensysteme ähneln den Management Information Systems (MIS) der USA der 1960er Jahre (vgl. Haigh 2001) und den Automatisierten Systemen der Information (ASU, Awtomatisierowannaja Sistema Uprawlenija), welche in der UdSSR als Typenlösungen im Einsatz waren (Peters 2017, 167f.).<sup>257</sup> Sie zeichnen sich durch Wiederverwendbarkeit innerhalb sich wiederholender organisatorischer Strukturen aus. Im Unterschied zur DDR zeichnet Peters für die UdSSR ein Bild, demnach die Einführung der ASU staatlich wenig koordiniert worden sei. In den USA handelte es sich ebenfalls um jeweils einzelstehende Lösungen, wobei BOMP durch die Marktmacht IBMs eine häufige Anwendung und Datenbank gewesen sein dürfte.

---

<sup>254</sup> Die Dissertation zitierte aus dem Entwurf der Direktive, veröffentlicht in der SED-Parteizeitung Neues Deutschland vom 5.5.1971.

<sup>255</sup> Andere Branchen waren nicht Gegenstand der Untersuchung. Die Studie merkte aber an, dass Materialwirtschaft, Planung und Abrechnung der Arbeitskräfte, Grundmittel, Kontokorrent und Informationssysteme auch in Handel und Verwaltung eingesetzt werden können.

<sup>256</sup> In Frage gestellt werden diese Prognosen allerdings durch den Zweckoptimismus, den die Dissertation im späteren Abschnitt 5.3 zur Weltstandsbestimmung postulierte (S. 157–159) und, so lässt sich zumindest retrospektiv feststellen, wieweit die ökonomische Dynamik in der BRD entweder grundlegend unterschätzte oder absichtlich ignorierte.

<sup>257</sup> Zu den ASU in der DDR siehe *Zur Herausbildung von Sichtweisen der Informatik in der DDR unter Einfluss der Kybernetik I. und II. Ordnung* (Fuchs-Kittowski 2007, 12).



Die DDR-relevante Innovation, die mit den SOPS gelang, waren Modularisierung und Automatisierung. Dies eröffnete die Möglichkeit, durch definierte Schnittstellen, Varianten von Softwaresystemen zu schaffen, die flexibel eingesetzt werden konnten. Eine wichtige Schnittstelle war dabei die Zentralisierung der Daten in einer Datenbank, wie BASTEI und später auf höherem Niveau DBS/R. Zu erkennen ist ein Lernprozess von generalisierenden Typenlösungen zu modularen Anwendungen, die für Sachgebiete miteinander kombinierbar sind. Wie in der DDR die staatliche Planung und Direktiven diese Entwicklung beschleunigten, konnte bereits aufgezeigt werden. Doch existierten in der DDR außerdem Prämissen der Digitalisierung und Vereinheitlichung, welche in der SOPS-Dissertation nicht explizit benannt wurden, und daher im nächsten Abschnitt herausgearbeitet werden sollen.

### 3.2.3.3 Einführung des SOPS BASTEI in drei Stationen

Wie dieser Digitalisierungs-Prozess mit Hilfe der netzwerkorientierten Datenbank BASTEI in den Jahren 1975–1977 von statten ging, ist an einer Reihe von Beispielen nachzuverfolgen. Die folgenden Beispiele stammen aus Veröffentlichungen in der Fachzeitschrift *rechentechnik/datenverarbeitung*. Sie unterliegen natürlich dem redaktionellen Filter, der an einer Problemdiskussion interessiert war, aber auch das Interesse verfolgt haben dürfte, für die Verwendung von SOPS zu werben. Sie sollen hier zusammengefasst wiedergegeben werden, um deren Anwendungsbreite als standardisierte Industrie-Softwarepakete zu verdeutlichen. Die Konzentration auf BASTEI erfolgt, um beispielhaft seine Nutzung zwischen Stücklistenverwaltung und Datenbank aufzuzeigen.

Erstens, benötigte das Chemiefaserwerk in Guben für seine Jahresplanung einen spezifischen Aufbau der Datenbank, da die chemischen Produkte auf Vorprodukten basierten, die wiederum auf unterschiedlichen Maschinen produziert wurden. Zielstellung war die Optimierung des Produktionsablaufes, so dass die Maschinen möglichst maximal ausgelastet waren. Im Unterschied zu den anderen Fallbeispielen wurde hier aus einer begrenzten Zahl von Ausgangsmaterialien eine große Zahl von Endprodukten durch Rekombination der Materialien erzeugt, die über verschiedene Maschinen zu verteilen waren (Ahrens 1976, 52). Das Problem verdeutlicht eine diagrammatische Übersicht (Abb. 59). Als Lösungsansatz wurde die Aufteilung in eine Produktdatei (welche die einzelnen Fertigungsergebnisse FE abbildet), eine Stücklistendatei (welche die Struktur abbildet, bzw. die für ein Produkt benötigten Zwischenprodukte Z), eine Arbeitsplatzdatei (welche die verfügbaren Maschinengruppen MG abbildet) und eine Arbeitsplandatei vorgeschlagen. Die Arbeitsplandatei verband die Produktdatei mit der Arbeitsplatzdatei und enthielt Informationen zum Leistungsvermögen der jeweiligen Maschinen-Gruppe, um auswählen zu können, welche Produkte darauf herstellbar waren.

Eine Auswertung der Arbeitsgangketten mithilfe von BASTEI ergab vorläufige Stücklisten, welche die tatsächliche Maschinenverteilung noch nicht berücksichtigten. Im nächsten Schritt waren für jedes Produkt qua Algorithmus optimale Maschinenlaufwege zu ermitteln, anhand dreier Eigenschaften je Maschine: Verarbeitungskapazität der Maschine, Bilanz aus Materialinput und Materialoutput und schließlich der maximal zu erzeugende Materialoutput. Indem ausgehend vom Endprodukt die Arbeitsschritte entlang der Maschinenkapazitäten verfolgt wurden, konnte letztendlich eine Stückliste erstellt werden, die alle Einzelpositionen beinhaltet. Für den zweiten Schritt wurde das SOPS OPSI (Optimierung mittels Simplextechnik) eingesetzt und damit modular mit BASTEI kombiniert. Das Gesamtsystem war so konzipiert, dass spätere Änderungen möglich waren, indem der Stapel erneut durchgerechnet wurde. Das Beispiel demonstrierte die Kombinationsmöglichkeit zweier SOPS, wobei

BASTEI vorrangig als Datenbanksystem verwendet wurde, und OPSI das Supply Chain Management prozessorientiert ermöglichte.

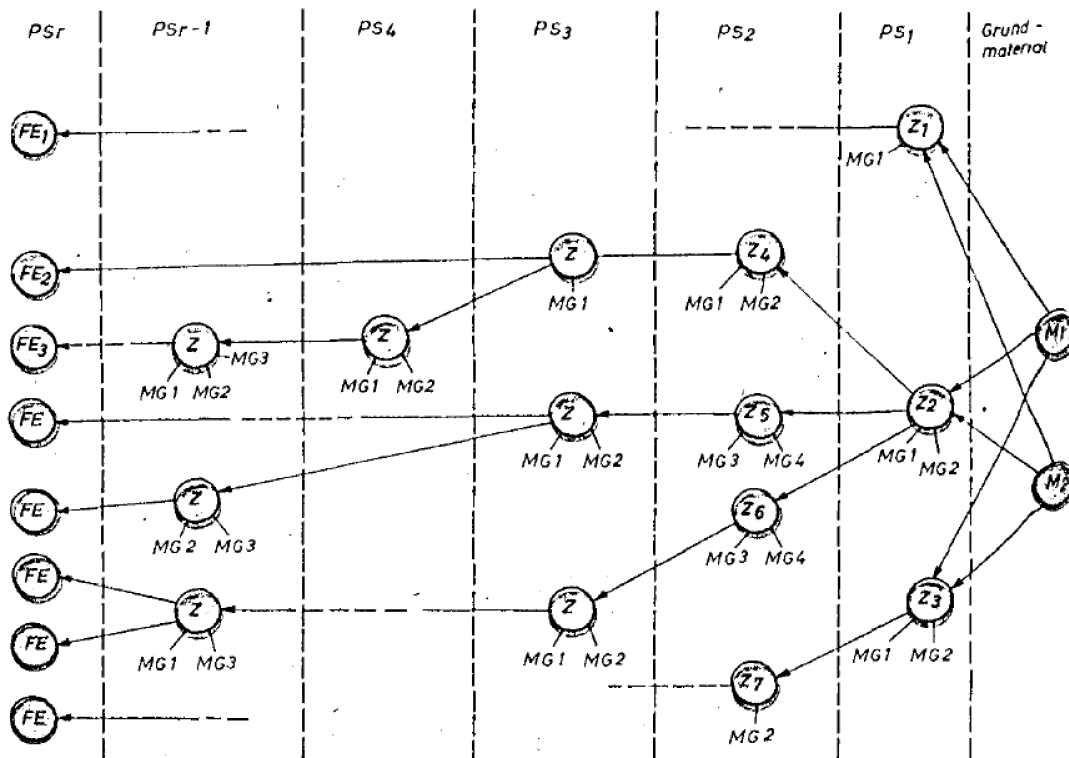


Abb. 59: Produktbaum in einem gerichteten Graphen – Kontraintuitiv von rechts nach links zu lesen. Rechts die Grundmaterialien und links die Endprodukte, dazwischen liegen verschiedene Produktionsstufen (PS). »Z« sind Zwischenprodukte und für jedes Zwischenprodukt ist markiert, welche der Maschinengruppen (MG1, MG2, MG3, MG4) zur Produktion verwendet werden können (Ahrens 1976, 52).

Zweitens, war die Erprobung von BASTEI im 1. Moskauer Uhrenwerk S. M. Kirow Teil des vereinbarten ESER-Projekts und sollte die Kompatibilität der von der DDR vertriebenen Komponente in der UdSSR überprüfen. Neben dem VEB Robotron war daran auch das ЦНИИТУ<sup>258</sup> aus Minsk beteiligt, welches in der UdSSR dem Ministerium für Gerätebau, Automatisierungsmittel und Leitungssysteme unterstand. Bis zu diesem Zeitpunkt verfügte das Uhrenwerk über eine teilweise Lochkartenverbuchung und über manuelle Verbuchung. Die BASTEI-Programmierer dirigierte damit den Auftakt zur Digitalisierung der Produktionsabläufe mittels elektronischer Computer. Zur Vorbereitung führten die Betriebsmitarbeiter in einem 12-monatigen Prozess für die gesamte Konstruktions- und technologische Dokumentation des Werkes neue Formblätter ein und vereinheitlichten die Bezeichnungen der Produkte, Baugruppen, Einzelteile und Materialien.

Neben der Installation von BASTEI und Erstellung der Datenbankstruktur testete das »Spezialistenkollektiv« die Abläufe anhand verschiedener Aufgaben: Erzeugung einer Gesamtstückliste aller Einzelteile, Auflösung der Gesamtstückliste nach Baugruppen und Produkten, Berechnung des Gesamtmaterialverbrauches und weitere. Dafür wurden Informationen aus 14 verschiedenen Primärdokumentation zusammengeführt, mit einem Datenvolumen von 70.000 Lochkarten. Im Ergebnis erhielten die

<sup>258</sup> ЦНИИТУ – Центральный научно-исследовательский институт техники управления. Staatliches Forschungs- und Entwicklungsinstitut für Steuerungstechnik. Vgl. <http://cniitu.by/history>.

Fachabteilungen Partituren aus 45 unterschiedlichen ausgedruckten Listen, welche die Arbeitsorganisation leiteten und unterstützten.

Das Rechenzentrum wurde mit einer Dokumentation ausgestattet, welche die weitere Nutzung nach Abzug des »Spezialistenkollektivs« ermöglichen sollte (Grunert 1975, 32). In der Auswertung errechnete der Autor für die 20-monatige Installation und Anpassung des Systems einen Aufwand von 10,1 Arbeitskräftejahren, die damit noch unter den durch Gräßler in der Dissertation veranschlagten 13,4 Arbeitskräftejahren lagen. Der trotzdem recht hohe Aufwand wurde einerseits auf die große Zahl zu integrierender Primärdokumentation zurückgeführt und andererseits auf die Notwendigkeit, die Programmführung, Fehlermeldungen und Dokumentation für die russische Sprache anzupassen. Der Autor stellte die Premiere von BASTEI in der UdSSR als positiv heraus, untermauert durch das Vorhaben, in den Jahren 1975 und 1976 weitere 80 Betriebe des Landes mit BASTEI auszustatten, sowie die Software zur Standarddatenbank des ЦНИИТ У Мinsk für weiter Planungssysteme zu erklären (ebd., 33). Die Erwartung und Vision, SOPS auch international, zumindest in den ESER-Mitgliedsländern verwenden zu können, habe sich erfüllt.

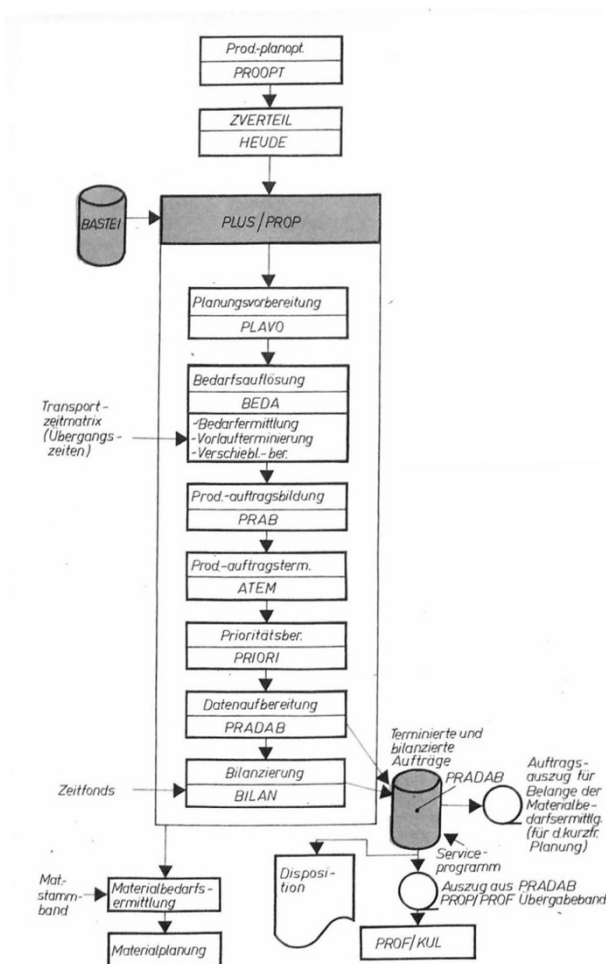


Abb. 60: Überblicksschema BASTEI Datenverarbeitung mit den PLUS-Teilmodulen (Apitz/Saffian/Koslowski 1976, 30).

Ein letztes, drittes, Praxisbeispiel ist die gemeinsame Entwicklung des VEB Dieselmotorenwerk Rostock und Institut für Schiffbau Rostock. Es wurde von 2500 Beschäftigten, an 400 Maschinen- und Handarbeitsplätzen betrieben. Jährlich stellte die Belegschaft durchschnittlich 50 Schiffsantriebe her, die aus Motor, Propeller, Steuerung, Werkzeugen und Ersatzteilen bestanden. »Je Motor sind etwa 4000-6000 Teile selber zu fertigen, wobei in einem Monat etwa 2500 Aufträge mit insgesamt 10.000 Arbeitsgängen anfallen« (Apitz/Saffian/ Koslowski 1976, 28). Bis zum Zeitpunkt der SOPS-Einführung gab es mehrere voneinander unabhängige Softwareprogramme zur mittelfristigen und kurzfristigen Planung.<sup>259</sup> Für die Neukonzipierung wurden die SOPS PLUS und BASTEI verwendet, wobei diese nicht in ihrer Gesamtheit, sondern als ineinandergreifende Modulbausteine Verwendung fanden (Abb. 60). Aus dem SOPS PLUS kamen zum Beispiel die Bausteine Produktionsplanung (PROP), Produktionsfeinplanung (PROF) und Kontrolle und Lenkung (KUL) zum Einsatz, in welche zusätzliche Eigenentwicklungen zur Planungsvorbereitung (PLAVO), Bedarfsauflösung (BEDA) oder Prioritätsberechnung (PRIORI) integriert

<sup>259</sup> Diese waren ursprünglich 1964 im Zuge des Neuen Ökonomischen Systems (NÖS) der Planung und Leitung installiert worden. Der VEB Schiffbau Rostock diente als Versuchsobjekt der NÖS und erhielt als einer der ersten DDR-Industriebetriebe einen Großrechner ZRA 1 (Bergien 2017, 277f.).

wurden. Das SOPS BASTEI diene als Datenbank. Die Teilkomponenten wurden einzeln programmiert und getestet, dann integriert und einem Komplextest unterzogen.<sup>260</sup> Die Probleme, die im Zuge der Einführung zu bewältigen waren, waren zahlreich: So fehlte es offensichtlich auf Leitungsebene an einer Zukunftskonzeption, welche Fragen der Planung, Vorbereitung, Durchführung und Abrechnung beinhaltet, und es gab trotz der erfolgreich abgeschlossenen Verträglichkeitsuntersuchung auf Leitungsebene Uneinigkeit in Bezug auf eine reine Nachnutzung der SOPS mit Erweiterungen vs. einer kompletten Eigenprojektierung. Zu vermuten ist, dass sich dieser Hinweis auf die Leitung des Betriebsrechenzentrums bezieht, auch wenn er im Text selbst äußerst neutral vorgetragen wird. »Den zügigen Fortgang der Arbeiten können die Leiter durch rechtzeitig und eindeutig gefällte Entscheidungen im Sinne der beschlossenen Konzeption beeinflussen« (Apitz/Saffian/Koslowski 1976, 32). Sowohl in der Leitung als auch in der Belegschaft war das Ziel der SOPS-Einführung nicht klar genug formuliert worden. Die Autoren fordern, dass sie »dem Neuen gegenüber aufgeschlossen gegenüberstehen, sowie bereit sein [sollen], sich weiter zu qualifizieren« (ebd.). Widerstände in der Belegschaft betrafen ein ganz wesentliches Problem der Automatisierung »formalisierbarer geistiger Prozesse« (ebd.): Mit der Übertragung an das Softwaresystem wurden den einzelnen Mitarbeitern Handlungsspielräume im Produktionsprozess genommen. Bis dahin implizites im Besitz der Mitarbeiter befindliches Wissen – ein potentieller Moment von Macht und Selbst-Ermächtigung – wurde expliziert und transparent gemacht, damit es in der Software modelliert und prozessualisiert werden konnte. »Die Mitarbeiter sind zu überzeugen, daß die Nachnutzung von Programmen keine minderwertige, geistlose Beschäftigung ist, und sie nicht überflüssig macht, um ihre volle Mitarbeit von Anfang an zu gewinnen« (ebd.). Wie das gelang, zeigt sich in der darauffolgenden Aussage, dass die Einführung des »qualitativ hochwertigen EDV-Projektierungs-Materials« zu einem »enormen Qualifizierungseffekt« (ebd.) führte. Dieser betraf vor allem Fragen der Organisation und der Rechentechnik, d.h. die Qualifizierung des mittleren Managements und der Mitarbeiter des Rechenzentrums.

Gleichzeitig zur Einführung der SOPS BASTEI und PLUS zentralisierte der Betrieb die Datenverwaltung und überarbeitete die Primärdokumentation, »was sich bereits vor Einführung des neuen Planungssystems positiv auf viele vorbereitende und durchführende Abteilungen auswirkt« (ebd., 33.). Projektgruppe und die Leitung kommunizierten die Abläufe für die Datenerhebung in Form von Arbeitsanweisungen in die Belegschaft. Die Verwendung vorgefertigter Softwaremodule und die damit einhergehende Zentralisierung und Standardisierung führten jedoch zu Kollisionen in den Zielvorgaben immer dann, wenn bestimmte Prozesse zu sehr von den Standards abwichen und damit nicht anpassbar waren. Angesichts dessen riefen die Autoren dazu auf, Ausnahmeregelungen »möglichst auf ›Normalfälle‹ zu transformieren« (ebd.) und in den Ablaufplan der Produktion einzubeziehen, das heißt Ausnahmen planbar zu machen oder in ihre Einzelteile zu zerlegen und somit tayloristisch zu betrachten.

Trotz dieser Widerstände und Probleme veranschlagten die Autoren die erreichte Zeiteinsparung in der Softwareentwicklung gegenüber einer Eigenentwicklung auf 60–80 Prozent. Insgesamt führte die Einführung eines »qualitativ höheren Planungssystems« mit Hilfe der SOPS in eine neue Qualität der Leitung und Planung des Betriebes, so die Autoren.

Probleme und Widerstände bei der Einführung der SOPS ergaben sich nicht allein zwischen Betrieben, ihren Belegschaften, den Einführungsteams und dem Hersteller Robotron. Auch innerhalb von Robotron war es schwierig, die Abteilungsleiter der SOPS von der Notwendigkeit einer datenbankgestützten

---

<sup>260</sup> Sie sollen hier zugunsten einer ausführlicheren Diskussion der im Artikel formulierten Erfahrungen mit der Einführung der Software nicht dargestellt werden, sind jedoch im Ausgangsartikel (Apitz/Saffian/Koslowski 1976) gut verständlich.

Datenverwaltung zu überzeugen. Die Gesamtkonzeption basierte auf Rolf Gräßlers betriebswirtschaftlicher Konzeption, welche jedoch nicht mit einer technologischen Konzeption, die beispielsweise genaue Schnittstellen zwischen den einzelnen SOPS definierte, unterlegt war. Dies erlaubte es, dass die für ein jeweiliges SOPS zuständigen Abteilungsleiter in ihren jeweiligen Abteilungen Abgrenzungen und Abkürzungen durchsetzen konnten, die der geplanten Integration teilweise zuwiderliefen.



Abb. 61: Importierter ESER-Großcomputer aus der UdSSR beim Fahrzeughersteller VEB Barkas-Werke zur Planung und Abrechnung sowie für Forschungsaufgaben (Wolfgang Thieme, Allgemeiner Deutscher Nachrichtendienst – Zentralbild, BARCH Bild 183-1983-1117-027).

Das 1973 bereits vor DBS/R fertiggestellte SOPS Grundmittelrechnung (GRUMI) besaß beispielsweise eigene Speicherstrukturen. Die Abteilungsleiter des Datenbankenbereiches bei Robotron konnten die Abteilungsleiter des SOPS GRUMI lange nicht von der Notwendigkeit einer Umstellung überzeugen. Auch das SOPS Materialwirtschaft wurde erst in einer späteren Version, ab 1975, mit der Möglichkeit ausgeliefert, es an die Datenbank DBS/R zu koppeln. Hier zeichnet sich ab, dass die spezifischen technologischen Kooperationen zwischen den Robotron-Abteilungen nicht, wie zu erwarten wäre, durch Weisungsgebung hergestellt wurden, sondern auf interpersoneller Ebene ausgehandelt wurden. Einer der Gründe war, dass der Datenbankenbereich und die Arbeit an DBS/R im längerfristiger Zeitrahmen operierte, währenddessen aus Wirtschaftlichkeitsgründen die einzelnen SOPS oft an schnellen Umsetzungen und Inbetriebnahmen interessiert waren. Außerdem war durch den Gesamtleiter Rolf Gräßler der Bereich Datenbanken als eines von vielen SOPS konzipiert worden und nicht als zentrales Datenreservoir. »Bei uns [bei Robotron] war jeder Abteilungsleiter sein König. Wir haben quer gekämpft, wir wollten unsere Technologie den anderen aufdrücken, aber die haben sich dagegen gewehrt. Die Führung war nie in der Lage, das wirklich zu steuern. Das ist der eine Grund: Dass ein Führungsmechanismus dort einfach gefehlt hat, dort einfach nicht stark war. Da war die Demokratie viel größer als man annimmt« (Interview Bittner/Heinemann 2018, Fol 17). Ein weiteres maßgebliches Problem für die in Datenbanken zentralisierte Datenhaltung war in den gesamten 1970er Jahren die hochkomplexe Programmierung, die im obigen Beispiel für BASTEI bereits angedeutet wurde. Das Wissen, um

netzwerkorientierte oder hierarchische Datenbanksysteme aufzubauen, breitete sich nur langsam aus – eine Entwicklung die der Robotron-Datenbankentwickler Jürgen Bittner als belegbar für die Situation in Ost und West ansieht (ebd.).

Ein wesentliches Problem in der DDR war nach Aussagen der Robotron-Mitarbeiter die unzureichende Ausstattung mit Hardware in den 1970er Jahren. Erst mit dem ESER-Rechner EC 1055 ab 1978 war eine ausreichend große Rechnerkapazität vorhanden, um beispielsweise SOPS und Datenbanken gleichzeitig bei zufriedenstellender Reaktionszeit zu betreiben (ebd., Fol 11; siehe auch Abb. 61). An diesem Punkt zeigen sich die Auswirkungen des CONCOM-Wirtschaftsembargos, welches den Export von IBM-Hardware in die RGW-Länder unterband. Die an IBM orientierten Eigenentwicklungen des ESER Systems in den RGW-Ländern waren mit einem Nachlauf von drei bis fünf Jahren in einer ähnlichen Leistungsklasse verfügbar, eine Situation die sich mit Einführung von Mikroprozessoren weiter verschlechterte, auf einen geschätzten Abstand von 7 bis 10 Jahren. Allerdings sind derartige Einschätzungen mit Warnzeichen zu versehen, denn sie orientieren sich mit der BRD an einer äußerst prosperierenden gesamtwirtschaftlichen Lage, die weltweit nur wenige Länder überhaupt betraf. Das Gros der Länder agierte unterhalb dieses hohen Niveaus und im Vergleich dazu mag die Hardware-Ausrüstung der DDR gar nicht so schlecht ausgesehen haben.<sup>261</sup> Derartige Aussagen sind mitunter auch der Ungeduld der Ingenieure zuzuschreiben, und die lässt sich nicht nur für die östliche Robotron beobachten, sondern auch für die westliche SAP, wie ein Beispiel um die Computermesse CEBIT 1991 in Hannover zeigen wird.

Faktisch sind jedoch die fehlenden Investivmittel in den DDR-Betrieben. Die daraus folgende unzureichende Hardwareausstattung der DDR-Betriebe und Organisationen war auch ein Grund dafür, warum Robotron lange Zeit am netzwerk-orientierten Datenbanksystem DBS/R festhielt. Relationale Systeme wurden in den 1970er Jahren in der DDR als zu ressourcen-intensiv angesehen, um über die Forschung hinaus, sinnvoll verwendet werden zu können.

Nachdem in diesem Abschnitt die Entwicklung der SOPS in der DDR ausführlich dargestellt wurde, gilt es nun, den Blick in den Westen zu richten.

### 3.2.4 Walldorf: SAP und die modulare Integration

In Westdeutschland entwickelte die Firma SAP in den 1970er Jahren eine Software, welche den SOPS in vielerlei Hinsicht ähnelte und sich doch davon unterschied. SAP ist heute zu einem globalen Player betrieblicher Software geworden. Doch soll hier nicht diese Erfolgsgeschichte erzählt werden, sondern ein möglichst detaillierter Bericht über die Datenbankintegration in frühen SAP-Produkten aus medienhistorischer Perspektive gegeben werden.

---

<sup>261</sup> Die Einschätzung der Wirtschaftsleistung der DDR ist komplex und je nach Forschungsstand mehrfach revidiert worden. Siehe beispielsweise die Aussage zu den methodischen Problemen der westdeutschen Wirtschaftsforschung bei der Erfassung der ostdeutschen Wirtschaftsleistung von Dr. Doris Cornelsen, bis 1993 Leiterin der Abteilung ›DDR und Osteuropa‹ am Deutschen Institut für Wirtschaftsforschung: »Die Weltbank stellte in ihrem bekannten Weltbankatlas bis Ende der 70er Jahre eine Rangfolge sämtlicher Länder auf, bei der die DDR einen der vordersten Plätze einnahm. Die Weltbank gab dann, weil es einige Diskussionen über das Problem dieser Eingliederung der Ostblockländer mit nichtkonvertierbaren Währungen gegeben hatte, eine große Untersuchung in Auftrag. [...] Diese Untersuchung brachte eine derartige Menge von methodischen Problemen zutage, daß die Weltbank schon Anfang der 80er Jahre damit aufhörte, diese Rang- und Reihenfolge in der Weise weiterzuführen und u. a. nahm sie auch die DDR als zehntgrößte Industrienation nicht mehr auf« (Cornelsen In: Enquete-kommission Überwindung der Folgen der SED-Diktatur im Prozeß der deutschen Einheit 1997, 186).

Zum derzeitigen Forschungsstand: Die Frühgeschichte des Produktes SAP, ein Akronym für ›Systemanalyse und Programmentwicklung‹, ist bisher wenig dokumentiert, da sich ein Firmenarchiv erst seit 2019 überhaupt im Aufbau befindet und sich die bisherigen Publikationen auf wenige verfügbare Quellen bezogen. Hauptprobleme der Geschichtsschreibung sind die relativ späte Wahrnehmung von SAP in bundesdeutschen Medien, die erst ab Ende der 1980er von der Firma überhaupt Notiz nahmen, und der Umstand, dass sich SAP selbst als ein zukunftsgerichtetes Unternehmen definiert, welches nur begrenzt historische Bedürfnisse habe (Leimbach 2007, 33–35). Ebenfalls problematisch ist der Umstand, dass Bibliotheken die Publikationen über frühe Systeme durch aktuellere Bücher ersetzt haben und eine nach wie vor zu konstatierende defizitäre Auseinandersetzung mit der Geschichte der Software in Deutschland. Der folgende Abschnitt referiert eine Reihe von Datenbanktechnologien, die zwar Spezialistinnen geläufig sind, jedoch kaum breiter bekannt sind, und die in der medientheoretischen Forschung bisher wenig Beachtung fanden. Er kann daher stellenweise unzugänglich wirken. Doch verbirgt sich hinter dieser Darstellung die Absicht zu einer Präzisierung, welche Grundlage für weitere Überlegungen medientheoretische Überlegungen sein kann. Ziel ist es darzustellen, wie die betrieblichen Bedürfnisse der Informationsverarbeitung eine spezifische Figuration aus Datenbanken und Enterprise-Resource-Management-Software hervorgebracht haben.

Eine einführende populäre, journalistische Geschichtsschreibung der Firmengeschichte von SAP findet sich in *SAP – die heimliche Softwaremacht* (Meissner 1997) und *Matrix der Welt – SAP und der neue globale Kapitalismus* (Siegele/Zepelin 2009). Der Historiker Martin Campbell-Kelly gab eine kurze Einordnung über SAP als Softwarefirma und die Genese von ERP-Software in der Monographie *From Airline Reservations to Sonic the Hedgehog – A History of the Software Industry* (Campbell-Kelly 2003, 191–197). Im Zuge seiner Dissertation *Die Geschichte der Softwarebranche in Deutschland – Entwicklung und Anwendung von Informations- und Kommunikationstechnologie zwischen den 1950ern und heute* stellte Timo Leimbach das frühe SAP angereichert durch Interviews mit frühen SAP-Mitarbeitern, erstmals im akademischen Kontext dar (Leimbach 2009, 287–295, 393–401).

Die Gründer von SAP, Hasso Plattner und Dietmar Hopp, arbeiteten 1970 von Mannheim aus für IBM als Projektprogrammierer. Initial für ihren Weggang von IBM war ein Projekt, welches sie für den Fiber-Hersteller Imperial Chemical Industries entwickelten. Das Bestellsystem sollte über 50 verschiedene Produktkombinationen ermöglichen, und Terminal-Eingaben in Echtzeit verarbeiten, damit die Kunden zu Veränderungen ihrer Bestellungen unmittelbares Feedback erhalten konnten, zum Beispiel zur Lieferzeit oder den Gesamtpreis betreffend. Mit dem Anspruch der Echtzeit-Kommunikation zwischen Terminal und Großcomputer griffen sie den damaligen medialen Wandel auf, der im Vergleich zu dem durch Systemoperatoren abgeschirmte Modus der Stapelverarbeitung einen direkteren Zugriff auf Computerressourcen erlaubte

Die Arbeitsziele dafür und die Vision einer integrierten Organisation dank integrierter Informationsflüsse hatten sie 1972 in einer IBM-Publikation unter dem Titel *Auftragsabwicklung, Disposition und Versandsteuerung integriert im Realzeitbetrieb* publiziert.<sup>262</sup> Darin werden die Motivation des Auftraggebers Imperial Chemical Industries deutlich, welcher aus Kostengründen keine hohen Lagerbestände vorhalten wollte. Daher müsse es »möglich sein, durch eine Dispositions- und Verfügbarkeitsrechnung frühzeitig Unterdeckungen zu erkennen, um rechtzeitig Gegenmaßnahmen treffen zu können«

---

<sup>262</sup> Die Broschüre war offensichtlich auch in der DDR verfügbar. Ein Exemplar wurde beispielsweise durch die Sektion für Rechentechnik und Datenverarbeitung der Technischen Hochschule Karl-Marx-Stadt (heute: Chemnitz) 1974 in die Bibliothek aufgenommen.

(Meier/Hopp/Plattner 1972, 6). Ein weiterer Schwerpunkt der Automatisierung lag auf der Ersetzung menschlicher Arbeitskraft. So sollten sich »die Sachbearbeiter auf Ausnahmesituationen konzentrieren können, während Routinearbeiten schneller, billiger und sicherer durch Maschinen abgewickelt werden« (ebd. 5). An dem bisherigen manuellen Informationsverfahren schätzten sie die hohe Flexibilität. Es weise allerdings auch eine ganze Reihe von Nachteilen auf: »nur geringe Auskunftsbereitschaft; unzureichende Planungsunterlagen; keine mit erträglichem Aufwand durchführbare Verfügbarkeitsrechnung; hohe Fehlerrate; verlorene Aufträge/fehlende Rechnungen; hoher Personalbedarf; Schwierigkeiten bei Personalwechsel« (ebd.). Für das maschinisierte Stapelverarbeitungsverfahren konstatierten sie, dass zwar diese Nachteile kompensiert werden konnten, allerdings die Flexibilität der Prozessorganisation stark abgenommen hatten. Sie schlugen daher ein »Realzeitsystem« vor, welches eine höhere Flexibilisierung bei gleichzeitiger Automatisierung ermöglichen sollte. Umgesetzt wurde es durch Bildschirmformulare, über die Werte einzugeben waren, also auch hier eine Lösungsvariante, welche Informationen aufnimmt und sich von der zeilenorientierten Befehlseingabe entfernt (Abb. 62). Rein informations-ästhetisch ähnelt diese Aufteilung bereits äußerst stark der Abbildung des Systems SAP R/2 aus den 1990er Jahren (Abb. 65). Mehrfach wurde in der Broschüre betont, dass die Einführung des Systems mit verstärkten Umstrukturierungen und Re-Organisation des Betriebsablaufes einhergehe. Letztlich habe dieses Vorgehen quantifizierbare Rationalisierungen ergeben, aber auch nicht-quantifizierbare, wie die Möglichkeit zum verbesserten Kundenservice (ebd. 27).

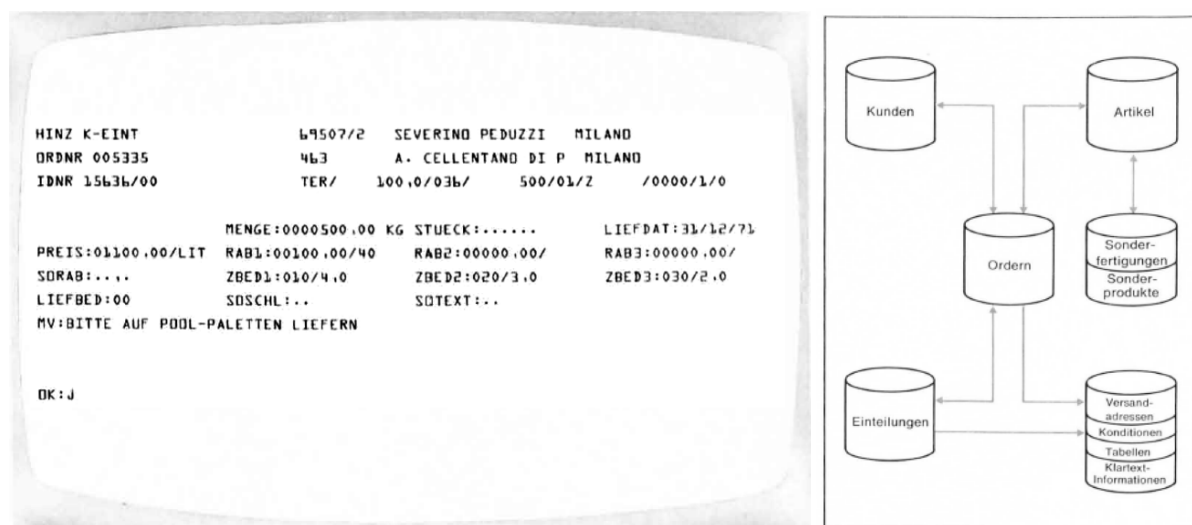


Abb. 62 (links): Eingabeformular mit Bestelldaten eines Auftrages. Daten wie Kundenadresse, kundenspezifische Lieferkonditionen und Produktinformationen werden aus der Datenbank »gezogen«. Bei Bedarf kann ein Teil der gezogenen Daten von den Sachbearbeitern überschrieben werden. Die Felder mit Punkten sind noch einzutragen (Meier/Hopp/Plattner 1972, 8).  
 Abb. 63 (rechts): Datenbank der Imperial Chemical Industries-Anwendung mit den Bestellungen, beziehungsweise »Ordern«, als zentrales Element (Meier/Hopp/Plattner 1972, 8).

Die Datenbank war als netzwerkorientiert im Direktzugriff (Direct Access) über Adressverkettungen konzipiert, ähnlich dem BOMP-Ansatz. »Eine Kette besteht aus mehreren Datensätzen (Gliedern) einer Datei, die durch Kettadressen miteinander verbunden sind. So wird in jedem Satz einer Kette die Adresse des vorherigen und des nachfolgenden Satzes festgehalten. [...] So bilden z. B. die zu einer Order gehörenden Einteilungen eine Kette, die im Ordersatz verankert ist« (ebd. 8). Auf diese Weise können über die »Ordern« beispielsweise Kunden und Artikel verknüpft werden oder auch die Kunden und Versandadressen. Deutlich wird, wie zentral die Transaktion in dieser Datenbank gesetzt ist (Abb. 13).



Das Vorhaben für Imperial Chemical Industries verließ den Bereich der reinen Anwendungsentwicklung. Immer deutlicher wurde, dass sich hier eine komplexe Software in der Entwicklung befand, die sich wiederverwenden ließ. Die dafür zuständige Produktentwicklung der IBM Deutschland war damals am Standort Stuttgart verortet. »Als Dietmar [Hopp] dieser Gruppe Fachfragen über das Verkaufsstatistik-System stellte, wurde klar, dass sie keinen Schimmer über ihre Vorhaben hatten, oder was sie bauen würden« (Interview Plattner 2012, Fol 2),<sup>clx</sup> so Plattner in einem Oral-History-Interview. Unzufrieden mit den Aussagen der Stuttgarter Kollegen verließen die Mannheimer Ingenieure Hopp und Plattner IBM, holten weitere IBM-Programmierer an Bord, und gründeten zusammen mit dem Physiker Klaus Tschira, dem Mathematiker Hans-Werner Hector, und dem Betriebswirt Claus Wellenreuther die Firma *Systemanalyse und Programmentwicklung*, kurz SAP.<sup>263</sup>

Sie setzten ihre Arbeit für Imperial Chemical fort, und hielten sich die Möglichkeit offen, eigene Software-Produkte auszukoppeln. 1973 stellte SAP die eigene Finanzbuchhaltung SAP RF<sup>264</sup> vor und zwei Jahre später die Materialwirtschafts-, und Instandhaltung als Modul SAP RM sowie 1980 mit SAP RV ein Modul für Vertrieb. Damit unterschied sich die Produktpalette vorteilhaft von IBMs Konkurrenzprodukt COPICS, dem eine integrierte Finanzbuchhaltung fehlte. Retrospektiv wurden diese Module durch als R/1 bezeichnet, zum damaligen Zeitpunkt als System R.<sup>265</sup> Eine gemeinsame, integrierte Datenverwaltung war in R/1 nicht vorgesehen.<sup>266</sup>

Zu den Motivationen dieses frühen Paketes, den Ideen und Hindernissen der Umsetzung äußerte sich Plattner 2012 in einem Interview: »Wir glaubten, dass die meisten Geschäftsprozesse, die wir kannten, vor allem die ganze Welt der Transaktionsverarbeitung, in Echtzeit machbar wäre. [...] Wir unterteilten die Stapelprozesse und ließen sie in Echtzeit ablaufen – in Quasi-Echtzeit, indem wir die Menge der Stapelverarbeitung reduzierten. Später führten wir doch Stapelanwendungen ein. Wir waren also noch nicht hundertprozentig erfolgreich. Doch die komplette Dateneingabe erfolgte zu 100% in Echtzeit, was zu einem System führte, in welchem, so sie denn erst einmal eingegeben waren, man sich auf die Daten verlassen konnte. Ein mehrstufiger Prozess um Daten zu validieren und zu akzeptieren wurde unnötig. [...] Das würde es für Geschäftsprozesse möglich machen, unmittelbar aufeinander zu folgen« (Interview Plattner 2012, Fol 4).<sup>clxi</sup> Aus dem Zitat geht deutlich hervor, wie in R/1 Batch- und Echtzeitprozesse ineinandergriffen, um bei den Nutzern den ›Eindruck‹ von Echtzeitzugriff zu vermitteln und es wird sichtbar, dass Daten den Ankerpunkt dieser Bemühungen darstellten. Damit stand die Frage der Datenverwaltung unmittelbar zur Debatte.

Mit dem Wissen um die ostdeutschen SOPS ist zu sehen, dass die Foki bereits zu diesem Zeitpunkt verschieden gelegt sind: Geht es laut Plattner für die SAP im Westen darum, Geschäftsprozesse in Echtzeit zu integrieren, also das Online-Potential von Computern auszuschöpfen, so lag die Motivation für die SOPS im Osten vorrangig in der betriebsinternen Rationalisierung und der Mehrfachverwendung

---

<sup>263</sup> Vgl. auch *Die Geschichte der Softwarebranche in Deutschland* (Leimbach 2009, 288f.).

<sup>264</sup> R stand für Realtime, dt. Echtzeit (Meissner 1997, 39). Rolf Gräßler veröffentlichte 2006 eine Tabelle, in der er die SOPS-Module den SAP-Modulen gegenüberstellte und aus der ersichtlich wird, dass beide in der frühen Periode über eine ähnliche topologische Ordnung verfügten (Gräßler 2006, 298). Dies ist jedoch kein Argument dafür, dass SAP von SOPS übernommen sei, wie Gräßler es gern verstanden wissen will, sondern dafür, dass trotz verschiedener politischer Systeme die Informationsbedürfnisse der Industrien weitestgehend ähnlich strukturiert waren.

<sup>265</sup> Campbell-Kelly verweist auf eine Reihe von anderen Firmen, die sich an den frühen SAP-Produkten orientierten: die Firma Baan aus den Niederlanden, und Systems Software Associates, PeopleSoft und J. D. Edwards aus den Vereinigten Staaten (Campbell-Kelly 2003, 172).

<sup>266</sup> Derzeit stehen keine weiteren Informationen zu den Details der R/1-Module zur Verfügung. Die frühesten für diese Studie verfügbaren Dokumente sind aus dem Jahr 1985. Das in Entstehung begriffene Archiv der SAP AG dürfte für weitere Forschung diesbezüglich relevant sein.

gleicher Module. Bezieht man die Motivationen mit ein, welche die Rationalisierung von Arbeitszeit in der IBM Broschüre zum Imperial Chemical Industries Projekt in den Vordergrund stellten, so spielten jedoch beide, SAP und SOPS, im Register der Rationalisierung.

Die Module von SAP R/1 wurden in der Folge bei zahlreichen Neukunden installiert und es entstand der Bedarf an weiteren standardisierten Modulen. Ebenso wie bei den SOPS handelte es sich bei R/1 nicht um eine fixe Softwarefiguration, sondern um das Zusammenspiel zahlreicher verschiedener Teile: Makros in Assemblersprache zur Datenverarbeitung und -aufbereitung, Code für Dateimanagement bzw. später auch Datenbankzugriff und eine Abfragesprache für Endkunden griffen ineinander mit einem Reportgenerator, der dazu diente, komplexe Berichte zu erstellen, die beispielsweise den Umsatz über einem bestimmten Zeitraum darstellen konnten. Die Firma konzentrierte sich auf die Entwicklung der Kernsoftware anhand exemplarischer Implementierungen bei Firmen. Waren die Module einmal generalisierbar entwickelt, gab SAP die konkrete Implementierung für die Anwendungsszenarien an Drittfirmen ab, welche Software- und Business-Consulting miteinander verbanden. SAP reduzierte damit das Risiko, selbst einen eigenen Stab an Beratern einstellen zu müssen (bei Robotron hießen sie Projektanten), kämpfte jedoch in Jahren des Booms damit, dass nicht genügend externe Berater zur Verfügung standen, die die Software bei den Firmen vor Ort anpassen und implementieren konnten. Dies hing auch mit der Komplexität der Software zusammen. Oft benötigte es jahrelangen Trainings, ehe SAP-Berater ausreichend in die Software eingearbeitet waren, um diese installieren und warten zu können.

Die Quellen zu den Datenhaltungsformaten sind komplex, denn in verschiedenen Technik-Konfigurationen wurden unterschiedliche Zugriffsmethoden verwendet und die derzeit verfügbaren Materialien über diese frühen Konfigurationen sind rar. So spricht ein ehemaliger SAP-Mitarbeiter in Bezug auf R/1 von: »Die Daten waren in Datenhaltungssystemen (DAM, ISAM VSAM) gespeichert« (Anon./Zipf 2019a),<sup>267</sup> und eine andere Quelle nannte außerdem »RFILE« (Anon./Zipf 2019b). Aus dieser Verschiedenartigkeit ergaben sich einerseits schnelle Zugriffszeiten, und andererseits komplizierte Konstrukte, welche einen hohen Pflegeaufwand nach sich zogen. Beispielsweise konnte ein SAP-System aus folgenden grundlegenden Dateien bestehen:

»ABEZ	Belegdatei zur Abspeicherung der Belege
ABMI	Beleg-Masterindex-Datei zur Verbindung der Belegnummern und der Belege in der Datei ABEZ
ABIB	Sekundärindex-Datei zur Verbindung der fortgeschriebenen Konten mit dem Beleg in der Datei ABEZ
SYSV	Nummernkreisdatei zur Vergabe der [fortlaufenden, F.H.] Belegnummer« (Wildensee 1997, Fol 41).

---

<sup>267</sup> DAM – Direct Access Method: Der Datensatz enthält eine ID, welche direkt, d.h. ohne Verwendung eines Index, auf die entsprechende Sektoradresse auf der Festplatte verweist. ISAM – Indexed Sequential Access Method: Mit Hilfe eines Index werden die Datensätze der Sektoradresse auf der Festplatte zugeordnet. VSAM – Virtual Sequential Access Method: Wie ISAM, jedoch ist der Index als sogenannter B-Tree (Balanced Tree) geordnet. Ein B-Tree ist ein Algorithmus, welcher die Schlüssel im Index ausgewogen verteilt. Beim Hinzufügen oder Entfernen von Schlüsseln stellt B-Tree diese Ausgewogenheit maximal wieder her, sodass alle mit den Schlüsseln referenzierten Speicheradressen mit gleicher Geschwindigkeit angesteuert werden. VSAM war Standard in vielen IBM-Großrechnern. RFILE – Ist ein Flatfile, indem die einzelnen Datensätze in eine Datei geschrieben werden und innerhalb der Datei durch Zeilen, welche als Datensatzheader dienen, voneinander getrennt sind.

Der Zugriff auf die einzelnen Belege erfolgte durch die oben genannten Zugriffsmethoden DAM, ISAM, VSAM und RFILE, wobei spezifische Programme die jeweilige Zugriffsmethode steuerten. Im Großen und Ganzen entsprechen diese Methoden den bereits im Kapitel 3.2.1.1 *Explosionen: Bankspeicherung Technischer Informationen (BASTEI) / Bill Of Materials Processor (BOMP)* besprochenen Zugriffswesen. Die Daten waren hierarchisch und maschinennah organisiert.

Ab Mitte der 1980er Jahre standen Änderungen im Raum: »Die Hauptmotivation für die Änderung der SAP-Basissoftware ist die Verringerung des Personal- und Maschinenaufwands bei der Modifikation der Anwendungssoftware« (SAP GmbH 1985a). Der Änderungsaufwand war bis dahin hoch, da jegliche Änderungen eine Neukompilierung des gesamten Systems nach sich zogen, was zu häufigem und langem Wartungsstillstand führte. Mit der Einführung und Weiterentwicklung von SAP R/2 sollte nicht nur dieses Ressourcenproblem bewältigt werden, es folgte auch eine Welle von Standardisierungsbemühungen, die nachfolgend aufgezeigt werden. Klaus Tschira fasste zusammen, SAP habe bis Ende der 1970er Jahre »[IBM] DOS, [IBM] OS und [Siemens] BS1000 unterstützt, unsere Daten in ISAM gespeichert und mittels CICS und ASMUS kommuniziert. Nun wollten wir auch mit VSAM, IMS, ADABAS und BS2000 (statt BS1000) in die 80er starten« (Tschira 1989, Fol 1).

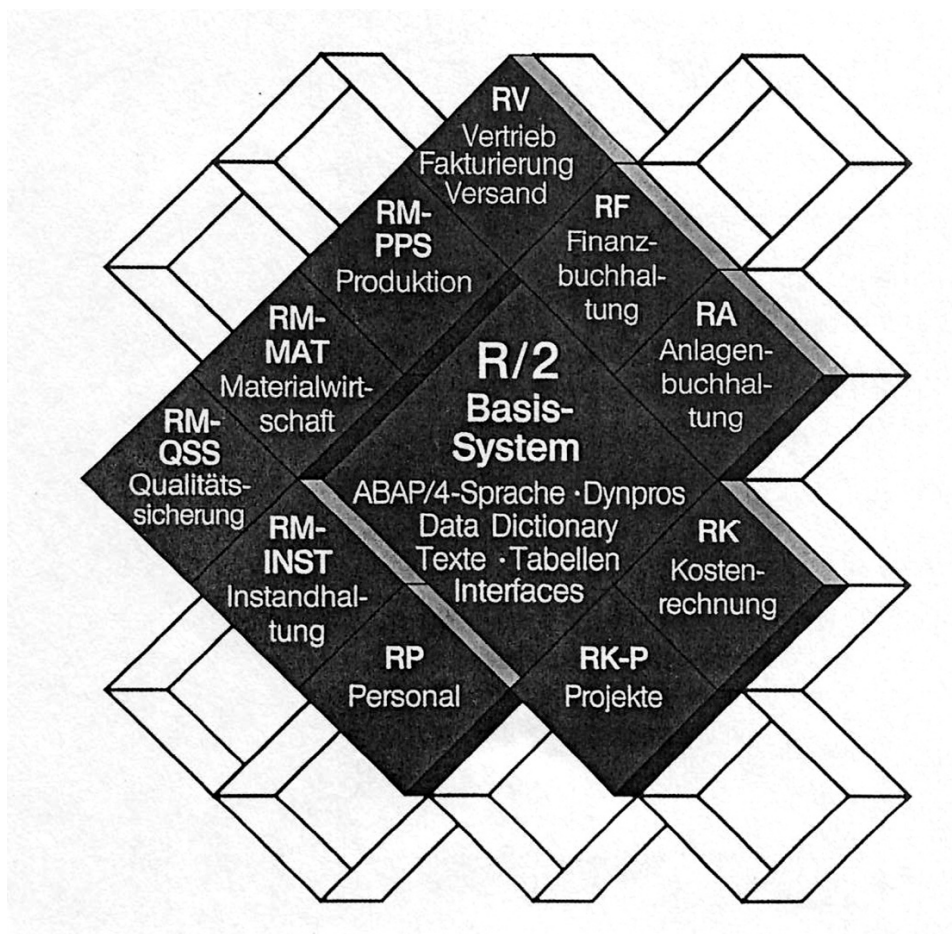


Abb. 64: Stand der Softwarefiguration SAP R/2 mit zehn Modulen Mitte der 1980er Jahre in einer Kundenbroschüre (SAP GmbH 1985b, 2)

Ab 1981 waren nach und nach die neuen Module Kostenrechnung, Produktionsplanung und -steuerung, und Personalwirtschaft eingeführt worden. Zusätzlich zu den bereits für R/1 genannten

Datenstrukturen und Tabellen, welche sich als langlebig erwiesen, schuf die Firma in R/2 nach und nach Anschlussmöglichkeiten an verschiedene Datenbanksysteme, zuerst die IBM IMS/DL1: »Die erste Version mit der Datenbank DL1 war die IMS/DL1 Version, die erstmals bei Freudenberg, Grundig, KHD und Mannesmann zum Einsatz kam« (Anon./Zipf 2019a). IMS/DL1 sollten bald durch weitere Technologien ergänzt werden, da sich die Netzwerklogik als problematisch erwies. Laut einer Kundeninformation aus dem Oktober 1985 entschied sich SAP in ihren Produkten dafür, nicht auf ein einzelnes Datenbankmanagementsystem (DBMS) zu setzen, sondern stattdessen Schnittstellen für verschiedene DBMS anzubieten: »Kriterien für die Verwendung von DB-Systemen sind ihre genügende Verbreitung und die Möglichkeit eine für operierende Systeme ausreichende Performance zu erreichen« (SAP GmbH 1985a, Fol 2). Neu sei eine Feld-orientierte Arbeitsweise. Die Daten wurden nicht mehr in sequentielle Dateien, sondern in einzelne Felder von Tabellen geschrieben. In der Broschüre orientierte die Firma ihre Kunden darauf, in Zukunft außerdem auf nicht-hierarchische Datenbanksysteme zu setzen, und nannte als Ziel das DBMS ADABAS.

Das Adaptable Database System (ADABAS) wurde ab 1969 als nicht-relationales DBMS für IBM-Großrechner durch die deutsche Firma Software AG entwickelt.<sup>268</sup> Das Datenmodell kam jenseits der relationalen Logik und des Netzwerkmodells ohne die Notwendigkeit zur Normalisierung aus. Die Indexe von ADABAS waren als invertierte Datei<sup>269</sup> abgelegt, das bedeutet, die verwaltenden Daten mussten nicht wie in relationalen Systemen strukturiert vorliegen. Zum Beispiel konnte ein Feld auch gemischte Datentypen enthalten. Im Unterschied zu netzwerkorientierten Datenbanken wie IMS/DL1, gab es keine Ketten von Pointern, die von einem zum nächsten Datensatz leiteten. Somit war ADABAS im Vergleich offener für Ergänzungen. Eine eingebaute Komprimierung ermöglichte zumindest in den Ressourcen-geplagten 1970er und 1980er Jahren Geschwindigkeitsgewinne gegenüber Konkurrenzprodukten. Als Abfragesprache fungierte das eigens entwickelte Adascript (Bowman 1975, 49), welches später um SQL erweitert wurde, sodass ADABAS auch relational bedienbar war.

Als relationales DBMS kam ab 1986 IBMs DB2, eine Nachfolgerin des IBM System R zur Anwendung (SAP GmbH 1985a, Fol 2).<sup>270</sup> Dessen Einbindung ab Mitte der 1980er Jahre war ein Zeichen der zunehmenden Produktreife relationaler Datenbankpakete, welche langsam dem Experimentalstadium entwachsen und neben den SAP-eigenen Tabellenstrukturen eine zunehmende Zentralstellung einnahmen. Die Zentralstellung von Datenbanken spiegelte sich auch in deren Prozessierungsbedarf wider: »Auf jeden Fall wird der Großteil der CPU-Zeit, welcher für SAP-Anwendungen benötigt wird, für das Datenbanksystem und den Transaktionsmonitor (CICS, IMS) verwendet« (SAP GmbH 1988, Fol 4).<sup>clxii</sup> Die Datenbanken waren mit den vorgefertigten SAP-Modulen zu verknüpfen. Da sich die Konfigurationen in verschiedenen Betrieben ähnelten, aber auch änderten, musste hierfür eine Anpassungsmöglichkeit geschaffen werden, welche ursprünglich durch eine Reihe von Assembler-Makros bereit gestellt

---

<sup>268</sup> ADABAS wurde laut Wikipedia wesentlich durch Peter Schnell entwickelt. Die Grundlagen dafür wurden in der Vorgängerin der Software AG gelegt, dem Institut für angewandte Informationsverarbeitung. Eine Oral History oder ähnliche Quellen sind bisher nicht verfügbar.

<sup>269</sup> Invertierte Indexe werden beim Einfügen von Daten oder Dokumenten in das DBMS angelegt und benötigen in diesem Moment Prozessorleistung. Sie vereinfachen Volltextsuchen und sind daher für Suchmaschinen-orientierte Anwendungen relevant. Für jedes Dokument werden relevante Wörter in eine Indexdatei aufgenommen. Diese besteht aus einer Liste aller indizierten Wörter aller Dokumente. Bei jedem einzelnen Wort sind die IDs der mit dem Wort verknüpften Dokumente aufgeführt, sodass bei einer Suchabfrage die Suchwörter rückverfolgt werden können zu den einzelnen Dokumenten. Dies ersparte Volltextsuchen, welche damals eine starke Last für Prozessoren und Peripherie darstellten.

<sup>270</sup> Vgl. *The History and Growth of IBM's DB2* (Haderle/Saracco 2013).

wurde. Ab R/2 Release 3 führte SAP im Zuge der Ablösung von Assembler<sup>271</sup> die besser verständliche Abfrage- und Datenmanipulationssprache *Allgemeiner-Berichts-Aufbereitungs-Prozessor* (ABAP/4) ein,<sup>272</sup> welche Ähnlichkeiten mit Hochsprachen wie beispielsweise COBOL aufwies. ABAP/4 diente der Erstellung der eigentlichen Software-Anwendung aus den Modulen. Darin wurden verschiedene Technologien zusammengefasst: »Mit den Macros (dynpro, PG, ABAP/3, DB-interfaces) verfügte SAP immer schon über eine eigene Programmierumgebung. Diese Funktionen, welche zuvor separat formalisiert waren, sind nun mit der ABAP/4-Sprache in einer vereinheitlichten Syntax und erweiterter Nutzbarkeit verfügbar« (SAP GmbH 1988, Fol 3).<sup>clxiii</sup> Während die Macros bei jeder Veränderung neu kompiliert werden mussten, konnte ABAP/4 zur Laufzeit interpretiert werden, wodurch ein flexiblerer, dialog-orientierter Umgang ermöglicht wurde. Unterstützt wurde diese neue Flexibilität durch in Ergänzung zu anderen Programmiersprachen SAP-spezifische Funktionen: Datums-Angaben, Währungsumrechnungen, Tabellenkalkulationen, Überprüfung von Zugangsberechtigungen und Datenoptimierung (ebd.).

Diese Funktionen können medien-genealogisch als spezifische Funktionalitäten gelesen werden, welche die Operationalisierung mit Datenbanken ermöglichten, vereinfachten und vereinheitlichten und damit andere Formen des Delegierens und Koordinierens verfügbar machten. ABAP/4 erhielt zudem Mitte der 1980er Jahre die Funktionalität eines Zugriffs auf relationale Datenbanken über eingebettete SQL-Abfragen, ein Zeichen dafür, dass derartige Funktionen nachgefragt wurden.

Nach dieser Beschreibung der grundlegenden Software-Figuration, gilt es, das Zusammenspiel der inneren Logiken mit den Interfaces zur Nutzung durch Endanwender nachzuzeichnen. Der Reportgenerator namens Logical Database, der jedoch im engeren Sinne kein DBMS darstellte, sondern hierarchisch organisierte Abfrage-Fähigkeiten zur Verfügung stellte, ermöglichte in R/1 und dem frühen R/2 die Erstellung von Berichten und Views.

Mit der Zeit kam die hierarchische Datenorganisation an ihre Grenzen (Anon./Zipf 2019b). Im Zuge der Einführung von ABAP/4 in R/2 Release 3 erfolgte 1988 eine Neuausrichtung, die im Sinne weiterer Standardisierung stattfand: Als Systemprogrammiersprache sollte fortan ›C‹ dienen, die Datenbanksprache auf SQL umgestellt werden, und es wurde auf das fensterorientierte Interface X/Windows (für Unix) und Microsoft-Windows orientiert (SAP GmbH 1988, Fol 3). 1989 veröffentlichte die Firma ein neues User-Interface für R/2 für die fenster-orientierte Bildschirmdarstellung (Abb. 65). Das Formular- und Tabellenorientierte User-Interface ermöglichte an Terminals eine visuell strukturierte Eingabe von Daten und erhöhte somit die Zugänglichkeit für Nicht-Spezialisten. Dazu zählte auch die Einführung von Menüs, welche zusätzlich zu den Eingabemöglichkeiten der Kommandozeile einen visuellen Zugang zur Programmstruktur schufen.

---

<sup>271</sup> Die Abwendung vom maschinennahen und schwer lesbaren Assembler-Code war nicht allein SAPs Strategie geschuldet, sondern ging auch von IBM aus, welche die Ablösung von Assembler durch die Programmiersprache C vorantrieb (SAP GmbH 1988). C konnte kompiliert und in Assembler übersetzt werden, und C-Compiler waren neben IBM-Rechnern auch auf denen anderer Hersteller verfügbar.

<sup>272</sup> Im Englischen wurde das Kürzel als Advanced Business (and) Application Programming übersetzt.

Pfleger Daten zur Person		ZEIT 10.17.57	DATUM 30.01.97
PERSNR.....	4711		
SAP Aktiengesellschaft			
WALLDORF			
GUELTIGKEIT	01.01.50 - 31.12.99	AENDG... 26.02.91	DMOSEK
		SPERR...	LKZ...
-----			
ANREDE.....	Herr		
NAME.....	Wanderer	GEB-NAME....	-
VORNAME....	Willy	INITIAL.....	---
VORSATZWORT.	-	NAMENSZUSATZ	-
AKAD.TITEL..	-		
-----			
ANSCHRIFT...	weg		
	68000 ma		
-----			
GEB-DATUM...	01.01.50 01.01.50	GEB-ORT.....	-
NATION.. D	deutsch	GEB-LAND ..	-
SPRACHE. D	deutsch		
FAMILIEN-ST.	ledig	SEIT.....	-
ANZ. KINDER.	-	KONFESSION..	-
			FC AE
OK	-		1 -50201

Abb. 65: Terminal-Eingabemaske für Personenstammdaten in SAP R/2 aus dem Jahr 1997. Die Navigation zu den Eingabefeldern erfolgte über Cursor und Tab-Taste. Eine Mausbedienung war nicht vorgesehen (SAP UEC (Anonymous) 2004).

Bis dahin erfolgte die Eingabe zeilenweise allein mit Kürzeln, sogenannten Transaktions- und Funktionscodes, die einen Bezug zwischen den eingegebenen Daten und dem Feld oder der Datei, in denen sie gespeichert wurden, herstellten, beziehungsweise das Wechseln in andere Programm-Kontexte ermöglichten. Das Kundenmagazin *SAP-Information* von 1985 schilderte die Neuerungen des Dialogbetriebs mithilfe von Fenstern und erwähnte folgende Aspekte: »Die Definition durch ASSEMBLER-Makros (IMAGE, SSTEP, EATAB) entfällt. Layout-Verschiebungen können vom Anwender selbst vorgenommen werden. Schlüsselwort und Feldformat werden automatisch aus dem Datenkatalog übernommen. Änderungen im zentralen Datenkatalog wirken sich sofort auf alle Bildschirmbilder aus« (SAP GmbH 1985a, Fol 3). Heute sind diese gravierenden Veränderungen der Nutzungsdimension größtenteils vergessen. Sie ermöglichten deutlich veränderte Umgangsweisen mit Computern.

Im Dezember 1989 blickte der Mitgründer Klaus Tschira in dem firmeninternen Rundschreiben *SAP in den 80er Jahren – Eine Betrachtung zum Dekadenwechsel von Klaus Tschira* auf das vergangene Jahrzehnt zurück. »Der neuen Generation der SAP-Anwendersoftware gaben wir den Namen R/2. Die mit dem Hardware-Generationswechsel von [IBM] /370 zu 43xx, von 77xx zu 75xx verbundene enorme Verbesserung des Preis/Leistungs-Verhältnisses bescherte uns einen schlagartig vergrößerten Markt für unsere Anwendungssysteme. Plötzlich hatten schon Firmen ab etwa 200 Mio. Umsatz genug Power, um unsere Systeme einzusetzen, und zusätzlich gab es ausreichend Luft für weitere Anwendungen« (Tschira 1989, Fol 1).<sup>273</sup> Insgesamt wertete Tschira die zurückliegende Entwicklung als »stürmisch« (ebd., Fol. 1), man sei von der Nachfrage »beinahe überrollt« (ebd., Fol. 2) worden. Diese Aussagen bezogen sich nicht allein auf das Firmenwachstum, sondern auch auf die spürbaren technologischen Veränderungen, welche sich durch Geschwindigkeits- und Kapazitätswachstum, die Emergenz neuer Speicherlogiken und beginnende Vernetzung auszeichneten.

Tschira gab zudem einen Ausblick auf die Entwicklungen der 1990er Jahre, der allerdings das sich abzeichnende Ende des sozialistischen Ostens noch nicht erwähnte. Gleichwohl entstand, mit der Ausbreitung der kapitalistischen Verwertungslogik in die östlichen Folgelandschaften quasi über Nacht ein neues Absatzgebiet für SAP, ebenso wie für den amerikanischen Datenbankgiganten Oracle und viele weitere. Zum Vorteil geriet, dass bereits in den 1980er Jahren die SAP Software ab der Version R/2 für

<sup>273</sup> Für die Abkürzungen vgl. Abb. 64.

den internationalen Einsatz konzipiert war. Sie ließ verschiedene Währungen und Zeichensysteme zu. Ab Mitte der 1980er Jahre war SAP in Europa, Afrika und Nordamerika vertreten (Campbell-Kelly 2003, 193; Leimbach 2007, 45).<sup>274</sup>

Tschiras Ausblick erwähnt genau diese internationalen Bestrebungen und konzentrierte sich anschließend auf die anstehenden technologischen Veränderungen: »Erstmals besteht dadurch die realistische Möglichkeit zur Implementierung einer einheitlichen Anwendungssoftware, die (1) auf Großrechnern, (2) auf mittleren Systemen wie der AS/400 oder größeren UNIX-Systemen und (3) auch auf intelligenten Workstations wie PS/2 und kleineren UNIX-Systemen ablauffähig ist« (Tschira 1989, Fol 3). Damit war der Weg zu einer stärkeren Dezentralisierung eröffnet, die den Produktnamen R/3 erhielt. Dazu gehörte auch die Möglichkeit verteilter Datenbanken.

Die Entwicklungen um R/3 sind besser dokumentiert als die frühen Ereignisse, und mit den beginnenden 1990er Jahren verlassen wir den Zeitraum der Untersuchung. Daher sei generell auf (Leimbach 2009, 393–401) verwiesen und nur einige Hinweise in Bezug auf Datenbanken gegeben. In einer Presseinformation von 1991 kündigte SAP an, R/2 vorerst weiter zu entwickeln, für jene Kunden, die nicht sofort auf R/3 umstellen wollten »bis für die dominierende Mehrzahl der SAP-Kunden R/3 einsatzfähig ist« (Tschira 1991, Fol 1). Neben dem bereits beschriebenen index-orientierten ADABAS (C) kam daher ab 1991 das voll-relationale Datenbankmanagementsystem ADABAS D zum Einsatz. Dieses wurde anfangs nicht durch SAP, sondern durch die Software AG betreut. ADABAS D geht zurück auf ein relationales Forschungsprojekt auf UNIX-Basis an der TU Berlin, Ende der 1970er Jahre.<sup>275</sup> Im Jahr 1997 übernahm SAP das relationale ADABAS D von der Software AG komplett und integrierte es unter dem Namen SAP MaxDB als relationale Datenbank in die eigene Produktlinie, wobei immer auch andere relationale DBMS, wie beispielsweise Oracle oder MySQL, anschlussfähig blieben.

Anfang der 1990er Jahre wurde die Umstellung auf relationale Datenbanken ein zunehmend wichtiges Thema für SAP R/3. Die relationalen Zugriffsweisen unterschieden sich von den bisherigen Vorgehensweisen. Die Firma versprach, ihre externen Berater gezielt zu schulen, da sie in dieser Umstellung den Hauptaufwand vermutete (ebd., Fol 2).

Zwischendurch geriet die Entwicklung von SAP R/3 ins Schlingern. Die Zentralkomponente des R/3 konnte nicht auf der damals vorgesehenen IBM AS/400 mit IBMs relationalen Produkt DB2 zum Laufen gebracht werden, da die Hardwarekapazitäten nicht ausreichten. Daher schwenkte die Entwicklung auf das Betriebssystem UNIX um. Doch ein wichtiger Test in der UNIX-Umgebung der VAX-Workstation scheiterte (Plattner 2012, 6f). Erst zwei Monate vor der Computermesse CEBIT, die im Januar 1991 in Hannover stattfand, eröffnete sich ein Lösungsweg. Kurzfristig gelang es, zumindest ein Softwaremodul, die Finanzbuchhaltung, auf einer Workstation zum Laufen zu bringen (Leimbach 2009, 399f.). Diese funktionierte im Zusammenspiel mit der relationalen Datenbank Oracle 6.2, welche auf den HP-Workstations 9000 unter UNIX lauffähig war und mit der Funktion Parallel Server den parallelen Betrieb von Datenbanken ermöglichte.<sup>276</sup>

---

<sup>274</sup> Zum Vergleich: Während die SAP auf verschiedenste internationale Märkte vordrang, kamen die ebenfalls mehrsprachig vorbereiteten SOPS außer in der DDR hauptsächlich nur in der UdSSR zum Einsatz.

<sup>275</sup> Rudolf Munz war einer der wesentlichen Entwickler des an der Professur H.-J. Schneider angesiedelten Forschungsprojektes *Verteilte Datenbanken Nixdorf* (VDN). Zu dieser frühen relationalen Forschung in West-Berlin fehlen weitere Angaben, die Desiderat bleiben. Dieses später DDB/4 genannte Datenbankmanagementsystem wurde durch die Nixdorf-Tochterfirma SQL Datenbanksysteme GmbH Berlin weiter entwickelt und mit der Lizenzierung an die Software AG zuerst in Entire SQL-DB-Server und später in ADABAS D umbenannt (Gfh 1991).

<sup>276</sup> Zur Firmengeschichte von Oracle siehe (Wilson/Mike 1997; Preger 2012; Mendelsohn 2013). Zur Genese der Konzepte Workstations und UNIX siehe (Ceruzzi 2003, 281–292).

Für eine längere Periode in den beginnenden 1990er Jahren setzte SAP somit auf zwei parallele Entwicklungen: Einerseits die Weiterentwicklung von R/2, unter anderem mit dem nicht-relationalen ADABAS (C) aber auch auf IBM-Rechnern lauffähigen relationalen Datenbanken wie IBM DB2 und auf der anderen Seite die Entwicklung von R/3 unter UNIX, welches allein auf relationale Datenbanken setzte, darunter vor allem Oracle und ADABAS D, später auch Informix und Microsoft SQL-Server. Die zugrundeliegenden Plattformen fächerten sich gegen Mitte der 1990er Jahre weiter auf, und R/3 wuchs aufgrund der erfolgreichen Client-Server-Konfiguration gerade in den USA stark.

Der Abschnitt zeigte auf, wie eng die Emergenz relationaler Datenbankmanagementsysteme verwoben ist mit der jenen Figuretionen, die heute als ERP (Enterprise Resource Planning) bezeichnet werden. Beide ineinander verwobene Softwarepraktiken zählen zum immens bedeutenden Feld der Logistical Media (Rossiter 2016). Es wird deutlich, dass neben der rein konzeptuellen Entwicklungen des relationalen Modells, wie in Kapitel 2.1 *Formatisierung – Strukturierte Daten in der relationalen Algebra* beschrieben, relationale Softwareanwendungen nach und nach in übergeordnete Softwarefiguretionen eingewoben wurden. Oft ermöglichten Datenbanken das Funktionieren von Betriebsorganisationssoftware überhaupt erst. Historisch gelangten vor den heute dominierenden relationalen Figuretionen der Datenorganisation andere, nicht-relationale Datenbanken zum Einsatz. Diese Vorgängerinnen sind insofern bedeutsam, als dass die Entwicklung relationaler Produkte aus den Schwierigkeiten im Umgang mit den vorhandenen Datenbankmanagementsystemen motiviert war. Auffallend ist die Ähnlichkeit der durch SAP entwickelten Module in R/2 mit den Modulen der Robotron SOPS. Die Gründe dafür werden im nächsten Abschnitt erkundet. An dieser Stelle setzt die zwischen ERP und Datenbanken verfügte Genealogie bei SAP im Westen aus, um den Blick erneut in den Osten zu lenken.

### 3.2.5 Dresden, Walldorf: Konzepte für die Formalisierung, Formatisierung und Informatisierung der Realität

In den 1980er und 1990er Jahren verdichteten sich die Konzepte von Anwendungssoftware für Verwaltung, Industrie und Organisationen, die aus standardisierten, anpassbaren Modulen bestanden. Im Folgenden soll der These nachgegangen werden, dass diese modularisierten Standardisierungen der Informationsverarbeitung ohne das Konzept der Datenbanken als Moment von Standardisierung nicht möglich gewesen wären. Dazu wird in einem ersten Schritt die Materialität und das infrastrukturelle Eingebettetsein von Enterprise Resource Planning-Software diskutiert. Im Unterschied zu den vorhergehenden Abschnitten, die in sich jeweils lose chronologisch organisiert waren, ist der folgende Abschnitt zugunsten einer materialorientierten und operationalen Darstellung chronologisch unscharf aufgebaut.

Das Akronym ERP wurde Anfang der 1990er Jahre durch die Beratungsfirma Gartner geprägt (Pollock/Williams 2009, 26). In den aus den Material Requirements Planning (MRP) hervorgegangenen Enterprise Resource Management (ERP) Systemen, unter die auch Customer Relationship Management (CRM) und Supply Chain Management (SCM) zu subsumieren sind,<sup>277</sup> treffen Standardisierung

---

<sup>277</sup> Eine Reihe paralleler Entwicklungen hat zu jenen Produkten geführt, die heute als ERP bezeichnet werden, so entstand parallel zum Material Requirements Planning auch die integrierenden Computer-Integrated Manufacturing (CIM) und Computer-Aided Production Management (CAPM) ab Mitte der 1980er Jahre, welche stärker noch versuchten Daten direkt über Sensoren und andere Gerätetechniken zu integrieren.



als Ausdruck von Generalisierung und die Anpassungsfähigkeit an lokale Gegebenheiten aufeinander. Mit der Soziologin Karen Ruhleder und der Informatikerin Susan Star kann Enterprise Resource Planning daher als infrastrukturelle Figuration bezeichnet werden, denn »Sie erwächst aus der Spannung zwischen lokalem, angepassten, intimen und flexiblen Gebrauch einerseits und der Notwendigkeit von Standards und Kontinuität« (Star/Ruhleder 1996, 114).<sup>clxiv</sup>

Durch die Verwendung parametrisierbarer Softwaremodule, sowie informationszentrierender Datenbanken und die Öffnung von Schnittstellen zu Zusatzsoftware (welche in der Lage ist, spezifische lokale Kontexte zu berücksichtigen), wird in ERP-Figurationen die infrastrukturelle Spannung zwischen globalen und lokalen Praktiken technologisch vermittelt. Während Material Requirements Planning sich auf den unmittelbaren Produktionsprozess konzentrierte, nimmt das daraus emergente ERP möglichst viele formalisierbare Aspekte wirtschaftlicher Tätigkeit auf. Es wurde nach Sachgebieten modularisiert, und mit Schnittstellen versehen.

Datenbank-MRP/ERP-Figurationen lassen sich damit in Abgrenzung zu reinen Material Requirements Planning-Systemen als Schwellentechnologie, die im Übergang vom Großrechner zu dezentralisierten, vernetzten, medial geprägten Nutzungsformen an Kleinrechnern und Personalcomputern entstand und einer dynamischen Standardisierung unterliegt, kennzeichnen. Eine Datenbank-ERP-Figuration ist in diesem Sinne nicht als Gesamtobjekt zu verstehen, sondern als miteinander in Beziehung gesetzte Teilelemente, welche selbst wiederum Verhältnisse und Beziehungen abbilden.

Frühe Datenbank-MRP/ERP-Figurationen der 1960er und der beginnenden 1970er Jahre konzentrierten sich auf spezifische Branchen. So notiert der Historiker Martin Campbell-Kelly für den Westen beispielsweise die Firma Hogan Systems als Herstellerin von Banken- und die Firma Policy Management Systems als Herstellerin von Versicherungssoftware (Campbell-Kelly 2003, 172). Erst mit COPICS, SOPS und SAP R/2 wurden hochintegrierte branchenübergreifende Lösungen entwickelt.

Sowohl die östlichen SOPS als auch die westlichen SAP R/2 Systeme können als generalisierte, integrierte Software, die in verschiedenen Branchen einsetzbar war, bezeichnet werden. Sie waren ausgerichtet auf kollaborative Prozesse und deren Steuerung und Kontrolle. Diese Figurationen aus einzelnen Modulen erlauben somit die ständige Herstellung, Erweiterung und Bewahrung von Beziehungen zwischen Prozessen, welche wiederum lokal, regional oder global verortet sind. Es sind Figurationen der Dynamisierung und Globalisierung von Geschäftsprozessen basierend auf der Vernetzung und Dynamisierung der zugrunde liegenden Kommunikation.

Wenn man SOPS und SAP R/2 als standardisierte Softwarepakete betrachtet – welcher Begriff von Standardisierung soll dann Verwendung finden? Die Medientheoretiker Geoffrey Bowker und Susan Star schlugen 1999 in ihrem Buch *Sorting Things Out* eine Reihe von Kriterien vor, die infrastrukturelle Standards ausmachen. Standards sind demnach vereinbarte Regeln, welche die Herstellung von Objekten ermöglichen. Ein Standard ist gekennzeichnet durch geographische und zeitliche Reichweite, die verschiedene Nutzer-Gemeinschaften und Nutzungsarten zu überbrücken vermag. Standards können ineinandergreifen, aufeinander aufbauen und einander überlagern. Sie sind zum Teil durch rechtliche Rahmungen abgesichert. Ein Standard muss nicht zwangsläufig die »beste« verfügbare Lösung sein, sondern kann in Aushandlungsprozessen auch durch eine Verhandlungspartei etabliert worden sein, welche über ausreichende Durchsetzungsmacht verfügt. Standards sind langlebig und können schwer und nur unter hohen Kosten geändert werden (Bowker/Star 1999, 13f.).

Am Beispiel der SOPS und ihrer Abstimmung auf die von der Staatlichen Zentralverwaltung für Statistik herausgegebenen Formblätter können wir einen mehrteiligen Standardisierungsvorgang

beobachten: Zuerst wurden (idealerweise) die organisatorischen Abläufe derart angepasst, dass sie mit den standardisierten Datenerfassungsbögen und Prozessablaufformularen integrierbar waren, damit in der Folge die standardisierten Softwaremodule mit möglichst wenig Anpassungen eingebunden werden konnten. Dabei ist die Vorgehensweise während der Einführung der standardisierten Formulare wesentlich: sie schafft bereits ein standardisiertes Modell der Wirklichkeit des abzubildenden Prozesses, indem im Formular die Entscheidung darüber getroffen wird, was überhaupt aufgezeichnet und zu Daten wird und welche Momente von Realität entfallen. Die Einhaltung dieser Standards erlaubt in der Folge das Aufsetzen standardisierter Software, denn die Realität wurde bereits in Formation gebracht und die formatierten Daten können maschinell verarbeitet werden.

Die zentrale Datenhaltung in Datenbanken stellt einen weiteren Moment von Standardisierung dar, denn sie kann nur erfolgen, wenn beispielsweise Vereinbarungen über die Verwendung gleicher Maßeinheiten an unterschiedlichen Eingabeorten vorliegen und so verschiedene lokale Gegebenheiten der Einsatzorte überbrückt und angeglichen werden. Die Datenzentralisierung und vor allem der online Zugriff transformieren aber auch die Bedeutung der Dateneingabe. Nicht mehr zeitverzögert, sondern unmittelbar haben Änderungen im Datenbestand Auswirkungen auf alle in Relation stehenden Nutzer. Die Ereignisdichte nimmt zu. Sie kann zu gefürchteten Verkettungen führen, weil eine falsche oder falsch interpretierte Eingabe, zum Beispiel die prozessierte Lagerhaltung und deren Vorhersagefunktion, für ein spezifisches Bauteil kaskadierend verfälschen kann.

Unabhängig davon, wo die Dateneingabe und -ausgabe erfolgte, bedeutete eine gemeinsame Datenbasis, dass die Verantwortung für jeden einzelnen Datensatz aus den Fachabteilungen übergangen an jene Abteilung, die das gemeinsame Datenreservoir zentral verwaltete, häufig also eine organisatorische Zentraleinheit, wie das Rechenzentrum. Diese wurde damit zu einem neuen Ort der Konstitution von Wissen, da bei Konflikten ihr die Daten- und Deutungshoheit zustand. Mit dem Medientheoretiker Mark Poster kann man in diesem Sinne Datenbank-Figurationen eine politische Wirkungsmacht zuschreiben, die gekennzeichnet ist durch eine bürokratisierte Form der Machtausübung (vgl. Poster 1995, 91f.).<sup>278</sup>

Das Vorhandensein von Standards zieht die Möglichkeiten der Veränderung und Ergänzung, aber auch des konfliktreichen Abweichens nach sich (vgl. Belke 1985). In komplexen infrastrukturellen Figurationen können sich diese Konflikte derart ausweiten, dass sie ein Scheitern, ein Nicht-Funktionieren und gegebenenfalls sogar den Rückbau verursachen. Zum Ausmaß der nötigen Anpassungen unter DDR-Bedingungen schätzte Wolfgang Schoppa, Professor an der Hochschule für Ökonomie, Berlin: »Standard-Software ist allgemein konzipiert. Damit entsteht vergleichsweise zur Eigenerstellung ein Anpassungsaufwand von 30 bis 50 Prozent« (Schoppa 1981, 7). Um den durch Standards gegebenen Verhärtungen entgegen zu wirken, bieten standardisierte Figurationen, wie beispielsweise die SOPS, eine Reihe veränderbarer Parameter. Das erlaubt es, innerhalb des Standards zu operieren, solange die Parameter breit genug ausgelegt sind und rechentechnisch trotzdem bewältigbar bleiben.

---

<sup>278</sup> Dieses Argument gilt es bei dezentralisiertem Datenbank-Gebrauch, welcher mit dem Einzug der Personalcomputer ab Anfang der 1980er zunimmt, zu kontextualisieren. Ab diesem Zeitraum sind mehrere Nutzungsweisen möglich: Die zentralisierte Nutzungsweise, Client-Server Architekturen, die teilzentralisiert sind und die individualisierte Nutzung auf einzelnen Personalcomputern.

Methoden dafür waren:

- die parametrische Konfiguration der Teilmodule,
- die Anpassung des Gesamt-Pakets,
- eine nur teilumfängliche Implementierung von Standardmodulen und die Ergänzung durch In-House programmierte individuelle Zusatzsoftware.

Als Voraussetzung für eine erfolgreiche Anpassung der Standardkonfiguration erwies sich, dass die verschiedenen Teilbereiche des technologischen Wissens und des organisatorischen Wissens miteinander in Beziehung gesetzt wurden. Diese Rekonfiguration der Ordnungen des Wissens innerhalb organisatorischer Einheiten ist hauptsächlich von erfolgreicher interpersoneller Kommunikation gekennzeichnet. Dem entgegen konnte die Ablehnung eines neuen Systems stehen, da je nach Position im Betrieb Wissens- und Machtkonstellationen, die auf der Kenntnis vorhandener Systeme beruhten, entwertet wurden und erst wieder neu aufgebaut werden mussten (Apitz/Saffian/Koslowski 1976, 33; Jenning 1979, 34).

Auch die Nutzer und Maintainer der Figurationen, die in den SOPS-Berichten größtenteils unerwähnt blieben, spielen eine wesentliche Rolle in erfolgreichen Rekonfigurationen der Ordnungen des Wissens: Durch interpersonelle Kommunikation – am technischen System vorbei – stellen sie das Funktionieren der in den SOPS materialisierten Standardisierungen sicher und überbrücken die Differenzen zwischen der modellierten Realität in den SOPS und der tatsächlichen betrieblichen Realität. Ohne diese häufig unsichtbar bleibende kommunikative Abfederung der in Code festgeschriebenen ERP-Datenbank-Figurationen, die sich außerhalb des Informations-Modells bewegen, das heißt, ohne den Möglichkeitsraum interpersoneller und informeller Absprachen, wäre ein Scheitern wahrscheinlicher. Durch die praktische Nutzung des ERP im Verhältnis zu den betrieblichen Anforderungen entsteht ein immaterieller Wissensraum. Wie lässt sich dieses Nutzungswissen situieren?

Wissen ist durch eine Reihe von Transformationen gekennzeichnet, welche Wissen erst erzeugen. In diesem Sinne ist das Wissen aus Transformationen der Nutzer und Maintainer anders situiert, als das der Planer, Entscheider oder der Programmierer. Letztere generieren neues implizites und explizites Wissen (Polanyi 1958) in Bezug auf die organisatorische und technologische Konstellation, während die Nutzer vor allem implizites Nutzungswissen erarbeiten. Werden die Nutzer ihres Wissens enteignet, wenn dieses durch Planer und Programmierer aufgegriffen und formalisiert wird? Auf der Erscheinungsebene ist durchaus ein Wissenstransfer aus den Fachabteilungen und von den einzelnen Beschäftigten erkennbar, der vor allem im Zuge der Modellierung der Informationsrealität anzusiedeln ist. Die Formalisierung von implizitem Wissen macht es transparent. Die Wissensträger geben demnach das bisher an die Person gebundene Prozesswissen weiter, indem es formalisiert und in Modellen, Algorithmen und Daten niedergeschrieben wird.

Dies muss allerdings, so argumentieren die Ökonomen und Organisationswissenschaftler Paul S. Adler und Bryan Boris, kein allein negativer Effekt sein. Sie halten negative Einstellungen gegenüber Formalisierung für ein Problem, welches entsteht, wenn die Arbeitsanforderungen und die organisatorische Umgebung einander nicht ausreichend angepasst sind. »Angestellte reagieren positiv, wenn ein hoher Grad an Formalisierung verbunden mit Routineaufgaben ist und wenn ein niedriger Grad an Formalisierung mit Aufgaben verbunden ist, die keine Routine darstellen. Wenn jedoch Arbeit zu stark formalisiert ist für eine anstehende Aufgabe – wenn zu viele Abläufe zu streng definiert sind – werden alle von den Kritikern von Formalisierung antizipierten Folgen erwartbar sein« (Adler/Borys 1996, 65).<sup>clxv</sup> Was Adler/Borys für die 1990er Jahre notieren, wurde auch in den 1970er Jahren in der DDR

beobachtet: Aus einer Studie der Sektion 04 Sozialistische Betriebswirtschaft der TU Dresden zitierend, konstatiert der Betriebswirtschaftler Matthias Jenning die Erkenntnis, dass »die Tätigkeiten mit einem hohen, rein routinemäßigen Arbeitszeitaufwand auch am stärksten durch die Anwendung der EDV unterstützt werden« (Jenning 1979, 32).

Für die DDR wurde argumentiert, dass die Transformation impliziten Wissens in formalisierte Systeme unproblematisch sei, da keine Gefahr von Arbeitslosigkeit bestehe. So formulierte der Abteilungsleiter im Ministerium für Wissenschaft und Technik, Günther Bretschneider, Ziele für die Ausarbeitung des Fünfjahresplanes 1975–1980. In Bezug auf den Arbeitskräfteeinsatz formulierte Bretschneider als Ziel die: »Erhöhung des gesellschaftlichen Arbeitsvermögens durch Freisetzung von Arbeitskräften für andere Aufgaben, durch Einsparung von Arbeitszeit für Routinearbeiten zugunsten der Arbeitszeit für schöpferische Arbeit sowie die Verhinderung des Anwachsens des Arbeitskräftebedarfs für eine Reihe von Aufgaben der Leitung und Planung sowie des Dienstleistungssektors. [...] die Übertragung zeitaufwendiger monotoner und physisch belastender Routinearbeiten der Steuerung und Überwachung von Produktionsprozessen an die Informationsverarbeitungstechnik oder die Entlastung der Mitarbeiter in Planung, Abrechnung und Verwaltung von Spitzenbelastungen und Überstunden« (Bretschneider 1974, 5). Aus retrospektiver Sicht des ostdeutschen Physikers und Historikers Gerhard Barkleit, war jedoch ein Teil dieser Argumentation nur vorgeschoben, da es vor allem um die Sicherung der Exportfähigkeit von DDR-Produkten zur Erwirtschaftung der dringend benötigten Auslandsdevisen gegangen sei (Barkleit 2000, 42).

Somit verlaufen die Rekonfigurationen der Ordnungen des Wissens in den DDR-Betrieben und der Grundlagenforschung entlang dreier Vektoren:

- Arbeitsintensivierung durch Identifizierung vorhandener Freiräume,
- Arbeitserleichterungen durch Reduzierung monotoner Tätigkeiten,
- Notwendigkeit zur Erhöhung der Produktion, besonders für den Export.

Diese standen im Spannungsfeld verschiedener, hierarchisch angelegter, diskursiver Praxen: der gouvernementale Diskurs durch SED und Ministerien, der informatorische Diskurs der Technologen und Maintainer und Nutzungsdiskurs der Anwender.

Für die SOPS der 1970er in der DDR konnten in der Literatur keine Beispiele gefunden werden, in denen das Standardisierungsvorhaben offensichtlich gescheitert ist. Das bedeutet nicht, dass alle Projekte erfolgreich waren. Es wurde wahrscheinlich über ein Scheitern weniger offen publiziert, als in den westlichen Ländern. In Erinnerung sei der Bericht über das VEB Dieselmotorenwerk Rostock gerufen, demnach sowohl in der Leitung als auch in der Belegschaft die Einführung des SOPS nicht klar genug mit Zielvorstellungen verknüpft worden war. Auch der Nachnutzungsauftrag und somit der Umfang, in dem die Standardlösung SOPS angepasst werden sollte, war im Vergleich zu einer kompletten Eigenlösung in Rostock anfangs umstritten.

Die Motivationen für Widerstände gegenüber der Einführung von SOPS und ERP konnten vielfältig sein. Aus der UdSSR berichtet der Historiker Ben Peters über einige gescheiterte Einführungen der teils ähnlich gelagerten Automatisierten Systemen der Leitung (ASU – Awtomatisierowannaja Sistema Upravlenija).<sup>279</sup> Diese konnten am Unwillen des mittleren Managements scheitern, welches die eigenen

---

<sup>279</sup> Upravlenija (in ASU) kann übersetzt werden als Leitung, Steuerung, oder Regelkreis und ist nicht nur sprachlich, sondern auch in der Genese in der frühen sowjetischen Kybernetik verankert (vgl. Берг/Китов//Ляпунов 1959). Die unter dem Kürzel zusammengefassten Systeme konnten sehr verschiedener Natur sein und orientierten in ihren Ursprüngen eher in Richtung von Management-Informationssystemen. Später wurden zunehmend pragmatische, produktionsorientierte ASU, versehen mit dem Zusatz »P« für Produktion, daher ASUP, in den Vordergrund der Entwicklung gestellt. Aus den vorliegenden

Informationsprivilegien schwinden sah, oder schlichtweg zu alt war, um sich die neuen, kybernetisch inspirierten Mittel anzueignen. Auch wurden ASU durch die Leitung boykottiert, da sie deren Fehler aufdecken könnten, so der Proponent Wiktor M. Gluschkow in der Tageszeitung Prawda (Peters 2017, 146).

Wie ein Scheitern der ERP-Implementation aussehen kann, wurde im Westen vor allem in den 1990er Jahren zahlreich dokumentiert. So berichtet der Betriebswirtschaftsprofessor und Analyst Tom Davenport von einem Versuch, 1992 bei Eastman Kodak ein globales Operations- und Finanzmanagement einzuführen, welches aufgrund des Widerstands von Managern, insbesondere des Chief Financial Officers, scheiterte. In einem anderen Fall eines Konsumgüterherstellers wurde nach einer sechsmonatigen, sehr spezifischen Reorganisationsplanung für das Bestellsystem und die Supply Chain SAP als ERP eingeführt und ein Großteil der ursprünglichen Planung zugunsten der »monströsen« Anforderungen des Paketes überschrieben (Davenport 1996, Fol 4f.). Der Analyst David Sprott verzeichnet für eine spätere Phase von ERP, in denen ältere bereits existierende ERP durch neuere Generationen abgelöst wurden: »Für Modulprojekte ist es ein ärgerlicher Umstand, dass ein erheblicher Teil von Implementierungen in den 1990ern die Zeit- und Kostenplanung sprengte. Diese Überschreitung war oft darin begründet, dass die Module mit anderen Anwendungsumgebungen integriert werden mussten. Das war normalerweise kein technisches Problem, sondern den Differenzen in Semantiken und Geschäftsregeln zwischen verschiedenen Anwendungen geschuldet, die ursprünglich nicht darauf angelegt waren, zusammenzuwirken« (Sprott 2000, 66).<sup>clxvi</sup> Er kennzeichnet damit die Erweiterung standardisierter Software und Einbindung in bereits vorhandene Routinen und Standards als problematisches Feld. Dass die Problematik der eingebetteten Standards umso virulenter wird, je mehr die vorhandenen ERP-Figurationen in die Jahre kommen, zeigen die eingangs erwähnten Beispiele vom Versandhaus Otto bis zur Firma Liquy Moly.

Die Robotron SOPS wurden auch deshalb ein Standard in der DDR, weil Robotron für Geräte und Softwareproduktion auf Großrechnern eine monopolartige Zentralstellung hatte, die nur von wenigen anderen Herstellern, wie den Elektro-Apparate-Werken Berlin und dem Leitzentrum für Anwendungsforschung Berlin ergänzt wurden. Durch die frühzeitige Entscheidung auf Ebene des SED-Politbüros Ende der 1960er Jahre wurden Aushandlungsprozesse in dem Sinne verkürzt, als dass nicht mehrere Hersteller um die Vormacht im Standardisierungsprozess kämpfen mussten, da es mit Robotron faktisch nur einen wesentlichen Hersteller gab. Zwar kamen als Softwareproduzenten die Rechenzentren der Bezirke und der Betriebe, die auf eigene Veranlassung und Kosten handelten, in Frage. Abhängig war deren Initiative allerdings vom kreativen Umgang mit der, durch die Plankommission zu genehmigenden, Jahresplanung. Ab der Zentralisierungsoffensive von 1969 standen umfangreichen Eigenprojekten entsprechende Gesetzgebung und Verfügungen zur Nachnutzung von Software entgegen. In dieser Hinsicht definierten das SZS und die SOPS-Entwicklungsabteilungen bei Robotron, was die *Best Practice* für standardisierte Softwareprodukte war. Ein Unterschied zu den westlichen Vorgehensweisen fällt unmittelbar auf: Das von IBM 1974 herausgegebene COPICS stellte als Best-Practice-Sammlung zuerst eine Aufsatzsammlung über die betriebswirtschaftliche Organisation und Re-

---

Informationen ist zu schließen, dass sie den SOPS nicht ähnelten, da sie nicht auf Modularität und Wiederverwendbarkeit angelegt waren, sondern dass es sich um Typenprojekte, also branchenorientierte monolithische Systeme handelte. Der Historiker Ben Peters stellt dar, wie diese in das nicht realisierte Gesamtstaatliche Automatisierte System der Rechnungsführung und Informationsverarbeitung OGAS (ОГАС – Общегосударственная автоматизированная система учёта и обработки информации) imaginiert wurden (Peters 2017, 108f., 167f.).

Organisation von Informationsflüssen dar. Diese wurde mit entsprechender IBM-Software nach und nach unterfüttert. Die Dokumentation der SOPS hingegen bestand von Anfang an hauptsächlich aus technologischen Darstellungen und Anwendungsbeispielen der Softwarepakete, ohne näher auf ggf. notwendige Re-Organisation einzugehen. Ob dafür Handreichungen seitens des SZS, im Zuge der Einführung der einheitlichen Primärdokumentation bestanden, ist nicht bekannt. Nach derzeitigem Forschungsstand waren die Betriebe selbst zuständig, wobei durchaus Unterstützung aus den Sektionen der Sozialistischen Betriebswirtschaft an den Universitäten der DDR erfolgt sein mag.

Auffallend in der DDR-Kommunikation über innerbetriebliche Reorganisation ist das Primat der Einführung neuer Technologien. Organisatorische Veränderungen hingegen treten eher maskiert auf:

»Die integrierte rechentechnische Lösung fordert eine integrierte organisatorische Lösung, wobei die organisatorische von vornherein in zeitlich überlappter Form auszuarbeiten ist. Die Organisationslösung muß nicht nur die Einheit von Planung und Abrechnung, sondern auch den Zwangslauf von der Produktionsvorbereitung bis zur Durchführung beinhalten« (Apitz/Saffian/Koslowski 1976, 33). Für die Einführung der SOPS lässt sich daraus der vorläufige Schluss ziehen, dass die Einführung neuer Technologien bewusst in den Vordergrund des Diskurses – zumindest in dem Maße, wie er sich in *rechentechnik/datenverarbeitung* und den Softwaredokumentationen erschließt – gestellt wurde. Sie ergab eine legitimierende Erzählung, die Änderungen der organisatorischen Struktur begründen konnte, auch wenn in der Praxis der Ablauf umgekehrt war: Zuerst mussten idealerweise die organisatorischen Änderungen im Zuge der Primärdokumentation verändert und danach die ERP-Datenbank-Figurationen installiert und integriert werden.

Die Entwickler der einheitlichen Primärdokumentation nahmen für sich in Anspruch »etwa 90 Prozent der Forderungen aus der teileherstellenden und teilefügenden Industrie« abzudecken (Schildhauer 1981, 5), woraus sich ein hoher Standardisierungsgrad ableiten ließe, vorausgesetzt die Durchdringung der Betriebe mit diesem System war ähnlich hoch. Hier zeichnet sich jedoch ein anderes Bild ab: So gibt beispielsweise ein Autor zum SOPS Materialwirtschaft (MAWI) an, dass die von 1972 bis 1979 erfolgten 60 Installationen nur 12 bis 17% der Großbetriebe mit mehr als 1000 Beschäftigten überhaupt betreffen. »Gleichzeitig verdeutlichte diese Untersuchung auch, daß etwa 90% der Betriebe eigene Programme und 10% Programme anderer Betriebe nutzen. Das bedeutet, daß jeder SOPS-Nutzer auch noch eigene bzw. andere Programme anwendet« (Jenning 1979, 31). Dies korreliert auch damit, dass in einer Zeit, in der der Hauptspeicher eines Computers noch in Kilobyte angegeben wurde, für den Einsatz der SOPS gerade auch in Zusammenhang mit einer Datenbank bestimmte Hardwarekonfigurationen (die ESER-Großrechner EC 1040 und EC 1055) nötig waren, die in der DDR nicht in ausreichendem Maße zur Verfügung standen. Diese Divergenz verweist auf einen Standardisierungswillen einerseits, der durch staatliche Maßnahmen wie Gesetzgebung, Beschlüsse, Schulungen und Öffentlichkeitsarbeit flankiert wurde und auf die tatsächlichen Standardisierungsdurchdringung.

Den hohen Nutzungserwartungen stand ein hoher Bedarf an Hardware gegenüber, wodurch bis in die späten 1980er Jahre die Erwartungen zwangsläufig enttäuscht werden mussten. Die überhöhten Vorstellungen der Möglichkeiten durch Standardisierung in SOPS wurden in der DDR regelmäßig und vielfach geschürt. Beispielhaft sei hier der Sektorenleiter der Staatlichen Plankommission Günther Bretschneider zitiert, welcher Bezug nimmt auf den 8. Parteitag der SED im Juni 1971 und schreibt: »Das sozialistische Wirtschaftssystem ermöglicht u. E. auf dem Gebiet des Einsatzes der EDV gegenüber den kapitalistischen Industriestaaten insbesondere folgende Vorzüge zu realisieren: 1. die ungehemmte Nutzung dieser Ergebnisse der wissenschaftlich-technischen Revolution zum Nutzen aller Werktätigen und zur Erhöhung ihrer schöpferischen Leistungen, 2. Anwendung der EDV über den Bereich einzelner Wirtschaftseinheiten hinaus und im Maßstab der ganzen Volkswirtschaft,

3. volkswirtschaftlich einheitliche Systemlösungen und Datenverarbeitungsprojekte für wiederholt auftretende Aufgaben [...]« (Bretschneider 1973, 5). Diese Ziele und Hoffnungen repräsentierten in der DDR die offizielle Staatsdoktrin der späten 1960er Jahre. Niedergelegt waren sie in einer Reihe von Gesetzen und Ministerialbeschlüssen, eingebunden in den Fünfjahrplan 1971–1975. Sie gingen aus dem gleichen technokratischen Gremium hervor, welche auch die Zentralisierung der Forschung, Entwicklung und Produktion, verwirklicht im Kombinat Robotron, und die Umgestaltung der vorlaufenden Bildung und Forschung angestoßen hatten: Ab Mitte der 1960er Jahre konstituierte sich beim Ministerrat der DDR die Regierungskommission für Maschinelles Rechnen, besetzt mit 13 Männern aus Verwaltung, Bildung und Industrie. Diese bereitete die Parteitag- und Ministerialbeschlüsse vor (Interview Merkel 2018, Fol 1f.). Darin deuten sich Tiefe und Länge der Steuerung von Forschung und Entwicklung an, welche Teil der Standardisierungsbemühungen wurden. Die DDR schien in den 1960er Jahren sehr motiviert zu sein, eine derartige strategische Zukunftsplanung auf den Weg zu bringen. Ob die Gründe dafür allein in ökonomischen Motivationen lagen, oder auch andere Fragestellungen eine Rolle spielten, soll im nächsten Abschnitt befragt werden.

### 3.2.6 Dresden, Berlin(Ost): Von der fließbandorientierten zur transaktionsorientierten Rationalisierung

In Ost und West glichen sich die Motivationen für Rationalisierung und In-Formatisierung. So verwies die SED-Parteizeitung *Neues Deutschland* in der Ausgabe vom 8. Februar 1965 auf die Notwendigkeit der Rationalisierung: »[Ernst] Gallerach, Erster Stellvertreter des Werkdirektors des VEB Carl Zeiss Jena zitiert auf der 5. Plenartagung des ZK der SED einen sowjetischen Fachmann mit der Aussage, daß ›bei gleichbleibendem technischen Stand der Datenverarbeitung im Jahre 1980 etwa die gesamte Sowjetbevölkerung in der Sphäre der Planung, Verwaltung und Rechnungslegung tätig sein müsste« (zit. n. Sobeslavsky 1996, 69 FN 288). Diese in der ›östlichen‹ Parteizeitung beschriebene Tendenz traf systemübergreifend auf Industriegesellschaften zu, denn das Motiv der Planung ist in beiden anzutreffen: Im Osten zentralisiert als Fünfjahresplan, im Westen auf der betriebswirtschaftlichen Ebene der Einzelkapitalien. Zum Vergleich: Der Informatiker und Historiker Jos Peeters führte am Beispiel der kapitalistischen Niederlande aus, »Das tägliche Problem, Tausende von Transaktionen des Warentauschs (z.B. Preise, Lieferungen und Bestellungen) zu verwalten, wuchs linear mit den verwendeten Zahlen und Mengen. Sie hätten daher durch eine Erhöhung der Mitarbeiter und Ausstattung bewältigt werden können, doch Probleme in Planung und Controlling wachsen exponentiell zu den verwendeten Zahlen« (Peeters 2009, 58).<sup>clxvii</sup> Eine Ausweitung der elektronischen Datenverarbeitung und Überführung lebendiger Arbeit in Maschinen im Zuge von Algorithmisierung war, zumindest erscheint es heute so, unumgänglich. Angesichts dieser beiden Statements entsteht die Frage, welche Konzepte von ›Rationalisierung‹ im Osten und im Westen zum Einsatz kamen.

Dieser Abschnitt soll sich vorerst auf den ›Osten‹ konzentrieren: Der Focus auf Rationalisierung durch Softwareeinsatz entstammte in der DDR nicht den Konkurrenzverhältnissen, sondern dem Versuch die Strukturprobleme der sozialistischen Planwirtschaft zu kompensieren. Hier sollten Soft- und Hardware helfen, Ressourcen zu sparen, vor allem Arbeitskraft, aber auch Prozesse zu optimieren und verdichtete Informationen an Entscheider zu übermitteln. Wenn aus ihrer Binnenperspektive die sozialistischen Länder überhaupt einen Vorteil gegenüber dem »kapitalistischen Klassenfeind« hatten, dann war es die Möglichkeit der zentralen-Top-Down-Steuerung von Forschung und Entwicklung. Die vermuteten

Potentiale zeigt beispielhaft das Protokoll einer Sitzung des Politbüros der SED vom 28. März 1967, welches konstatierte, dass: »unter sozialistischen Produktionsverhältnissen wesentlich bessere Möglichkeiten für den Einsatz der Datenverarbeitung mit hohem ökonomischen Nutzeffekt gegeben sind als in den kapitalistischen Ländern« (zit. n. Sobeslavsky 1996, 114). Kurt Sack formuliert diese Zuversicht in seinem bereits zitierten Beitrag in *rechentechnik/datenverarbeitung* so: »In den kapitalistischen Ländern ist die Basis für eine breite Nachnutzung von anwendungsorientierter Software stark eingegrenzt. Das führt [...] zu rückläufigen Tendenzen auf dem Gebiet der Anwendersoftware. Diese Tendenzen wirken im sozialistischen Wirtschaftsgebiet nicht« (Sack 1974, 28). Sack adressierte den Punkt, dass im ›Westen‹ Software dezentral produziert werde, da es dort im Unterschied zur DDR keine zentralen Bezirksrechenzentren und keinen zentralen Akteur der Softwareproduktion wie Robotron gäbe. Daraus schloss er, dass die westliche Softwarekrise der 1970er Jahre (Ceruzzi 2003, 105–108) die DDR nicht treffen könne.

Wie verzahnte sich diese Zuversicht mit der Idee der Rationalisierung? Verschiedene Stimmen aus DDR-Publikationen riefen breit gestreute Anwendungsgebiete für die Rationalisierungsbemühungen auf: Im *Beschluß zur Erhöhung der Effektivität und zur Durchsetzung der sozialistischen Rationalisierung bei der Einsatzvorbereitung für die elektronische Datenverarbeitung* durch den Ministerrat der DDR hieß es 1971: »1.3. Die Datenverarbeitungsprojektierung ist umfassend zu rationalisieren, um die Projektierungszeiten spürbar zu verkürzen und den Projektierungsaufwand zu senken« (Ministerrat der DDR 1971, 523). Zwei Jahre später forderte G. Bretschneider, Sektorenleiter der Staatlichen Plankommission im Editorial der Zeitschrift *rechentechnik/datenverarbeitung* die weitere Rationalisierung und Vereinheitlichung: »Die Vorbereitung und der Einsatz der EDV ist aus den Erfordernissen rationeller Leitungssysteme abzuleiten und stärker als bisher mit der Rationalisierung und einer Vereinheitlichung der Betriebsorganisation zu verbinden« (Bretschneider 1973). Der Ökonom Hans Kunau vom Ministerium für Wissenschaft und Technik betonte drei Jahre später an gleicher Stelle: »Von besonderer Bedeutung ist dabei, überall dort einheitliche Datenverarbeitungsprojekte zu entwickeln und zu nutzen, wo das durch Vorhandensein oder die Schaffung einheitlicher Prozeßabläufe möglich ist« (Kunau 1976). Es müssten, so Kunau, »stärker die objektiv vorhandenen Möglichkeiten zur Vereinheitlichung, Typisierung bzw. Standardisierung von Organisationsabläufen und Prozessen, von Kennziffern, Normen und Ordnungssystemen, von Belegen und Dokumentationen ausgeschöpft werden« (ebd.). Ende der 1970er Jahre vermerkte ebenda Günter Zillmann, Stellvertreter des Ministers für Wissenschaft und Technik: »Gleichzeitig sollte die Rechentechnik [...] noch stärker als Beschleuniger des wissenschaftlich-technischen Fortschritts wirken, indem sie umfassend bei der Lösung wissenschaftlich-technischer Aufgaben, vor allem bei der Mechanisierung und Automatisierung von Routineprozessen in Forschung und Entwicklung genutzt wird« (Zillmann 1979, 6f.). In einer Diskussionsrunde zu Datenbankerfahrungen erhoffte sich Richard Kurth von der Staatlichen Zentralverwaltung für Statistik die zukünftige automatisierte Informationsvermittlung aus den Betrieben in die Planungsebene: »Man sollte mehr und mehr auch an den gesellschaftlichen Nutzen denken, der unter anderem daraus resultieren wird, daß die DDR zunehmend über Datenbanken verfügt, die größtenteils in irgendeiner Weise miteinander in Verbindung stehen. Es gibt Überlegungen, ob nicht die wachsende Zahl von Kombinationsdatenbanken dazu führen müßte, den Fluß der statistischen Informationen in der Volkswirtschaft zu verändern« (Kurth in: Loll u. a. 1982, 21). Und in einem Artikel, welcher zur Orientierung der Anwender von Datenbanken in der Fachzeitschrift *Neue Technik im Büro* 1985 erschien, rahmte Rolf Heinemann, Leiter der Robotron-Datenbankentwicklung Rationalisierung als Aufgabe des Koordinierens betrieblicher Abläufe: »Zur rationellen Beherrschung der unterschiedlichen betrieblichen Abläufe auf kommerziellem, technischem und technologischem Gebiet ist der Einsatz von Datenbanksystemen eine



international anerkannte Grundvoraussetzung. Fast jeder Betrieb und jede Institution, die eine integrierte Anwendung der Datenverarbeitung plant, ist nicht zuletzt auf den Einsatz eines Datenbankbetriebssystems angewiesen« (Heinemann 1985, 108). Warum war in den untersuchten Publikationen der ›sozialistischen‹ DDR die Frage der Rationalisierung derart prominent vertreten, dass sie als diskursleitend bezeichnet werden kann?<sup>280</sup>

In den Quellen werden eine ganze Reihe von Motivationen sichtbar, warum Rationalisierung für die DDR wichtig war: Der Projektierungsaufwand sollte gesenkt, die Betriebsorganisation und -leitung sollte vereinheitlicht und Prozessabläufe gestrafft werden. Routineaufgaben sollten maschinell ersetzt, Betriebsabläufe koordiniert und statistische Informationen direkt an die Behörden zur Wirtschaftsplanung übersendet werden. Es lohnt sich daher, den Diskurs der Rationalisierung und Automatisierung in der DDR näher in Augenschein zu nehmen, um zu umreißen, ob und wie er sich vom westlichen Diskurs unterschied.<sup>281</sup> Wie wurde der Rationalisierungsdiskurs in der DDR spezifisch gerahmt? Welche Bedeutung kam der Rationalisierung zur Bewältigung wirtschaftlicher Krisenerscheinungen zu? Auf welche Weise wurde in der späten DDR ein über Fließband-Analogien hinausweisendes Rationalisierungskonzept diskutiert, während in der Praxis Transaktionen immer wichtiger wurden? Welche Rolle spielte das Verhältnis aus informatorischen Alltagspraxen im Verhältnis zu kybernetischen Konzepten? Wie beeinflusste die zunehmende Wissensorientierung des Produzierens die DDR-Diskussion um Rationalisierung ab den 1980er Jahren? Wie begegnete der Rationalisierungsdiskurs schließlich dem Übergang von den vergleichsweise zentral organisierten Großrechneranlagen zu den dezentral verteilten Personalcomputern?

Erstens, zur ›genuin sozialistischen Produktivkraft Wissenschaft‹: Die Automatisierung im Realsozialismus wurde als eine Folge der Anwendung der Produktivkraft Wissenschaft diskutiert. Das Konzept setzte sich ab den 1960er Jahren zunehmend im DDR-Diskurs durch, um den Wandel zu wissensbasierten Produktionsverhältnissen und Reproduktionsweisen zu fassen. Eingebbracht den Diskurs wurde das auf Marx zurückzuführende Konzept 1957 durch den Architekten, Bauwissenschaftler und Staatssekretär Gerhard Kosel. Es dauerte einige Jahre bis es auch wirklich akzeptiert und verwendet

---

<sup>280</sup> Im Zeitraum zwischen 1973 und 1989 setzten sich in *rechentechnik/datenverarbeitung* inklusive der hier zitierten, mindestens 75 Beiträge unter Bezugnahme auf Software, Datenbanksysteme und SOPS mit Rationalisierungsfragen auseinander. Häufig wurde das Editorial der Zeitschrift verwendet, um strategische Aufforderungen zur Rationalisierung durch die Leitungsebene der DDR zu veröffentlichen, während in den Einzelbeiträgen Taktiken der Rationalisierung anhand von Fallbeispielen dargestellt wurden.

<sup>281</sup> Begriffsprägend im Westen war 1952 die Monographie des Wirtschaftswissenschaftlers und Entrepreneurs John Diebold: *Automation – The advent of the automatic factory*, welche u.a. kybernetisch inspiriertes Kapitel zu *Control and Computer* enthält. *The Re-design of Product and Process* sagte die zunehmende Kundenorientierung in der Produktpalette voraus und *The Automatic Handling of Information* formulierte die Hoffnung auf eine Reduzierung bürokratischer Tätigkeiten durch Automatisierung, um abschließend die sozioökonomischen Folgen der Automatisierung zu diskutieren (Diebold 1952).

Der Soziologe und Mitbegründer des Instituts für Sozialforschung in Frankfurt/Main Friedrich Pollock trug ab 1956 mit der soziologischen Studie *Automation – Materialien zur Beurteilung ihrer ökonomischen und sozialen Folgen* zur Debatte aus Perspektive der kritischen Theorie bei. In klassisch materialistischer Argumentation beschreibt er Automation als kybernetisiertes Ersetzen menschlicher Arbeitskraft »in der Werkstatt und im Büro durch Maschinen« (Pollock/Friedrich 1956, 2). Er plädierte dafür, diese neue »Produktionsweise« nicht der unkontrollierten Entfaltung durch Marktkräfte zu überlassen, sondern sie »unmittelbar und planmäßig für das Wohl der Menschen nutzbar zu machen« (ebd., 288). Mit dieser Begründung sprach er sich gegen Ende der ansonsten eher beobachtend gehaltenen Studie für eine »geplante Wirtschaft« aus, wobei eine Automatisierung des Informationsflusses helfen würde, die Planung in einer Komplexität zu bewältigen, wie es ohne elektronische Rechner bis dato nicht möglich gewesen sei (ebd., 289).

(vgl. Kosel 1957). Retrospektiv wertet der Technikhistoriker Joachim Radkau das von Marx entlehnte Konzept der ›Wissenschaft als unmittelbare Produktivkraft‹ in der DDR als »technizistische Ideologie« (Radkau 1990, 16). Es handele sich um die »einseitige Fixierung auf die Produktion unter Vernachlässigung der Reproduktion: auf diese Formel könnte man viele Fehler der DDR-Politik zurückführen [...]« (ebd., 24). Radkaus Argument zielt auf den bekannten Investitionsstau in der DDR ab: Instandhaltung und Erneuerung der Produktionsmittel wurden bereits im Plan nicht ausreichend berücksichtigt, sondern der Warenausstoß in den Vordergrund gestellt. An Gerhard Kosels Monographie fiel Radkau auf, diese argumentiere nicht mit der Bedürfnisbefriedigung der Menschen, sondern »vielmehr in der Mission, einen von der Wissenschaft gewiesenen Auftrag des revolutionären Fortschritts zu vollziehen« (ebd., 31). Diese Art von Denkfiguren sieht Radkau allerdings in anderer Form auch in der BRD am Werke.<sup>282</sup>

Zur Diskussion um die ›unmittelbare Produktivkraft Wissenschaft‹ äußerte sich ebenfalls der Technikhistoriker Rolf Sonnemann.<sup>283</sup> So zeigte er 1996 die diskursiven Konfliktlinien auf, die sich in der DDR durch eine ernsthafte Marx-Lektüre zwangsläufig ergaben: »Die Produktionsinstrumente, nach Marx das ›Knochen- und Muskelsystem der Produktion‹, sind ein Gradmesser für das Niveau der Produktivkräfte. Sie geben darüber Auskunft, wie weit sich der Mensch von seinem ursprünglichen Naturzustand entfernt hat. Daß sie auch ›Anzeiger der gesellschaftlichen Verhältnisse‹ sind, ›worin gearbeitet wird‹, hat im Laufe der Beschäftigung mit der Marxschen Theorie heftige Diskussionen ausgelöst. Es konnte schließlich niemandem entgehen, daß sowohl in der BRD, einem durch die Dominanz des Privateigentums an den Produktionsmitteln charakterisierten Land, als auch in der DDR, in der, zumindest formal, das gesellschaftliche Eigentum an Grund und Boden, Arbeitsmitteln und Arbeitsgegenständen herrschte, im Prinzip die gleichen Produktionsinstrumente angewandt wurden« (Sonnemann 1996, 9).<sup>284</sup> Nicht nur die Verwendung der gleichen Produktionsinstrumente war diskursiv prekär, auch der zunehmend sichtbare Rückstand der DDR gegenüber dem eigentlich als ›überholt‹ geltenden kapitalistischen System brachte qua Produktivkraftargument die DDR-Legitimation in Nöte.<sup>285</sup>

---

<sup>282</sup> Für die DDR konstatiert der Historiker Martin Sabrow, dass sich der Fortschrittsbegriff wandelte und zwar »von einem utopische Aufbruchskonzept mit totalitären Zügen in den fünfziger Jahren zu einer wissenschaftlich berechenbaren Plangröße in den sechziger Jahren, um dann in den siebziger Jahren mehr und mehr zu einer fremdbestimmten Herausforderung zu werden und schließlich mit seiner kulturellen Erschließungsfunktion insgesamt auch den Rest seiner legitimatorischen Bindungskraft zu verlieren« (Sabrow 2004, 184). In den 1960er Jahren sei der Fortschritt im Zuge der Kybernetisierung verwissenschaftlicht worden, verlor damit aber seine utopische Kraft, »umgeformt zu einem Vorwärtsschreiten, das als ›planmäßig, kontinuierlich, harmonisch, unerschöpflich‹ imaginiert wurde« (ebd., 178). Darauf folgte angesichts der immer reeller werdenden wirtschaftlichen Probleme der DDR ein langsames »Absterben der Zukunft« (ebd., 180), so Sabrow, den Historiker Rainer Gries (1994) zitierend. Schließlich trat der wissenschaftlich-technische Fortschritt der DDR als objektive Notwendigkeit gegenüber und zwang sie, so Wolfgang Heinrichs von der Akademie der Wissenschaften in der SED-Parteizeitung Neues Deutschland vom 13.7.1986, in einen »Wettlauf mit der Zeit« (zitiert n. ebd., 183).

<sup>283</sup> Rolf Sonnemann hatte gemeinsam mit Gisela Buchheim in der Sektion Philosophie und Kulturwissenschaften an der TU Dresden, Ende der 1970er Jahre, ein Forschungs- und Lehrprogramm zur Geschichte der Technikwissenschaften etabliert. Dieses ging aus dem Sektions-Bereich der Geschichte der Produktivkräfte hervor. Den Kontext schildert Thomas Hänseroth in einer Einleitung zum Wiederabdruck von Gisela Buchheims 1978 veröffentlichten Aufsatz *Zur Wechselwirkung von Naturwissenschaften und Technikwissenschaften in ihrer historischen Entwicklung*, welcher durch Hänseroth als »Gründungsschrift der Technikwissenschaftsgeschichte in Deutschland« markiert wird (Hänseroth 2010).

<sup>284</sup> Die Marx-Zitate sind durch Sonnemann angegeben aus dem 1. Band von *Das Kapital – Kritik der politischen Ökonomie. Der Produktionsprozeß des Kapitals* (Marx 1890, 23:195).

<sup>285</sup> Es war ein teils eigenverschuldetes Problem, da beispielsweise Walter Ulbricht mit der Devise ›Überholen ohne Einzuholen‹ die dogmatische und aus heutiger, kritischer Sicht falsche Lesweise der Marxschen Kritik der politischen Ökonomie, der Sozialismus müsse zwangsläufig produktiver sein als der Kapitalismus, perpetuiert hatte. Ulbricht gab vor, dass die Besserstellung

Kosel stützte sich auf eine Marx-Passage aus den ökonomischen Manuskripten von 1857/58, die seit 1953 in der DDR publiziert war: »Die Natur baut keine Maschinen, keine Lokomotiven, Eisenbahnen, [...] Sie sind *von der menschlichen Hand geschaffene Organe des menschlichen Hirns*; vergegenständlichte Wissenskraft. Die Entwicklung des *capital fixe*<sup>286</sup> zeigt an, bis zu welchem Grade das allgemeine gesellschaftliche Wissen, knowledge, zur unmittelbaren *Produktivkraft* geworden ist [...]« (Marx 1858, 42:602, Hervorh. im Original).<sup>287</sup> Doch wie so viele Marxisten unterlag Kosel dem Mißverständnis, bei Marx' Kapital handele es sich um eine positiv lesbare Anleitung, aus der ökonomische Schlüsse für eine sozialistische Gesellschaft zu ziehen seien.<sup>288</sup>

Die durch die SED gerahmte, ökonomisch-technologische Ideologieproduktion der DDR bekannte sich zu dem Konzept der ›unmittelbaren Produktivkraft Wissen‹ und formatierte damit den möglichen diskursiven Rahmen. Dabei verschob, so Laitko, das Konzept der ›geistigen Produktion‹ den Diskurs »von der Zirkulation (relativ) fertigen Wissens auf den Prozeß seiner Erzeugung« (Laitko 2005, 483). Es bleibt daher zu verfolgen, wie in der DDR der Prozess von Wissens›erzeugung‹ situiert war. Radkaus Verdikt über die einseitige Fixierung auf die Produktion unter Vernachlässigung der Reproduktion ist weiterhin prägnant für den DDR-Rationalisierungsdiskurs.

Zweitens, ›Rationalisierung als Krisenbewältigung‹: Das Einsetzen des Rationalisierungsdiskurses verortet Simon Donig in den frühen 1960er Jahren mit der Einführung des Neuen Ökonomischen Systems der Planung (NÖS) in Reaktion auf die Krisen im Zuge des vorangegangenen Fünfjahrplanes. »Kosteneffizienz und Rationalisierung wurden zu zentralen Schlagworten der Planer, mit Kybernetik und Computertechnik sollte die ökonomische Misere überwunden werden« (Donig 2009, 90).<sup>289</sup> Donig

---

gegenüber dem Westen in Bezug auf die Arbeitsproduktivität eine »Frage von Wissenschaft und Technik sowie der materiellen Interessiertheit« sei (Ulbricht am 18. November 1962, zitiert n.: Sabrow 2004, 48f).

<sup>286</sup> Zum ›capital fixe‹ zählen – in Abgrenzung zum variablen Kapitel, der menschlichen Arbeitskraft – in der Marxschen Theorie Rohstoffe, Renten (Miet- und Kapitaleinnahmen) und Maschinerie. Der Satz ließe sich somit paraphrasieren als: Das Entwicklungsniveau der Maschinerie zeigt an, in welchem Maße Wissen unmittelbare Produktivkraft geworden ist.

<sup>287</sup> Die Fortsetzung des Zitates lautet: »und daher die Bedingungen des gesellschaftlichen Lebensprozesses selbst unter die Kontrolle des general intellect gekommen und ihm gemäß umgeschaffen sind« (ebd.). Marx' Bemerkung der Maschinen als »Organe des menschlichen Hirns« resoniert mit der Theorie des Anthropologen André Leroi-Gourhan über die Auslagerung des Gedächtnisses in Form von Aufzeichnungen als Basis wissensbildender Überlieferung (Leroi-Gourhan 1964, 235; vgl. Frabetti 2014, 99ff.; vgl. Hartmann/Frank 2015, 108). In Anbindung an Gourhan manifestiert sich die vergegenständlichte Wissenskraft damit nicht allein im beschriebenen Papier, welches der Aufbewahrung von Gedachtem dient, sondern, folgt man den Ausführungen des Tabellenkapitels, in der Art und Weise wie das Gedachte zu Papier, bzw. zu Datenbank gebracht wird. Wenn auch sich die Studien Marx' auf die ihm vorliegende historische Situation Ende des 19. Jahrhunderts bezogen, und er die uns heute zur Verfügung stehenden symbolverarbeitenden Maschinen nicht gemeint haben konnte, so schließt sich doch via Exteriorisierung der Bogen zu den beschreibbaren Datenbanksystemen, welche in rechnenden Maschinen als »Organe des menschlichen Gehirns« sich materialisiert haben. Natürlich geschah dies nicht in der von Leroi-Gourhan antizipierten Form, dernach das Denken komplett in Maschinen ausgelagert wurde, sondern als kooperatives Wechselspiel zwischen humanen und non-humanen Akteuren, also als Verschiebung, nicht als Auslagerung. So argumentiert auch (Schüttpeitz 2006, 7).

<sup>288</sup> Dieses Zitat mit dem Ausgangspunkt des General Intellect war für die post-marxistische Strömung der Post-Operaiten zentrales Motiv und wurde beispielsweise auch durch (Terranova 2003) und (Virno 2008) und als Fixpunkt weiterer Überlegungen durch (Raunig 2015) verwendet. Kritisch zu post-operaistischen Lesweise äußert sich (Heinrich 1999, 349ff.).

<sup>289</sup> Zur Planungseuphorie und technokratischen Steuerung, deren Bezugspunkt in der »Zentralverwaltungswirtschaft« lag, merkt Hübner an, dass das Fehlen wirtschaftlicher Korrekturen durch Marktmechanismen zu einer Bürokratisierung führte, welche der Austarierung des Planes gewidmet war. Dieses habe in einer Entökonomisierung des Wirtschaftsverhaltens gemündet, und forderte, so Hübner, »technokratische Korrekturen geradezu heraus« (Hübner 1999, 326) und mündete schließlich in eine »Mischung technokratieähnlicher und improvisatorischer Handlungsmuster der Funktionseiliten«

legt nahe, dass die wiederkehrenden Rationalisierungskampagnen mit den ökonomischen Krisen der DDR zusammenhingen, eine These, die auch Barkleit im Hinblick auf Initiativen zur Mikroelektronik in der DDR bestärkt (Barkleit 2000, 17f.). Die Einführung von SOPS bedurfte großer Investitionen, die im Rahmen der Betriebe durch die Leitung verantwortet werden mussten, als Investitionsmittel in der Planung verankert wurden und durch die übergeordnete Plankommission zustimmungspflichtig waren.<sup>290</sup> Da ab 1960 auf allen Parteitage der SED die Informatisierung der DDR-Wirtschaft wieder und wieder als Zielvorgabe aktualisiert wurde, konnten die Investitionen jeweils begründet werden. Die Realität der Planwirtschaft bedeutete aber auch, dass die Investitionsmittel in anderen Bereichen fehlten, und das Fehlinvestitionen einschneidende volkswirtschaftliche Auswirkungen zeitigten. ›Rationalisierung‹ diente als Begründung, um den Ressourcenbedarf diskursiv zu lenken.

Der Glaube an das Eintreten der positiven Effekte der ›4. industriellen Revolution‹, scheint in Bezug auf die noch neue Elektronische Datenverarbeitung der späten 1960er Jahre nur ansatzweise mit Zahlen hinterlegt gewesen zu sein. Relativiert wird die positive Untersuchung der Wirtschaftlichkeit von SOPS von 1972 (Gräßler 1972) durch eine Studie des Betriebswirtschaftlers Dr. Matthias Jenning, welcher in der Zeitschrift *rechentechnik/datenverarbeitung* über *Die EDV in der Materialwirtschaft 1979* befand: »Zum Zeitpunkt der Projektanmeldung war kein Anwender in der Lage, den Nutzen für den Betrieb zu quantifizieren« (Jenning 1979, 32). Ob und in welchem Maße eine Effizienzsteigerung durch automatisierende Rationalisierung erfolgte, ließ sich also in Zahlen zu Projektbeginn nicht ausdrücken. Insgesamt scheinen die Betriebe allerdings gute Erfahrungen gemacht zu haben, denn, so Jenning weiter, es »wollen 86 Prozent der befragten Betriebe die EDV-Anwendung in der Materialwirtschaft erweitern, verbessern oder ändern. Mit den überwiegend positiven Erfahrungen wächst das Bedürfnis nach weiterer Rationalisierung der Leitung der materialtechnischen Versorgung« (ebd., 33).

Drittens, ›Langes Festhalten an der Tonnenideologie‹: Ursprünglich bezog sich Rationalisierung in der DDR auf die Skalierung industrieller Produktion, die »große Fallzahl« (Bretschneider 1973, siehe Zitat oben). Kritisch wurde die Orientierung an großen Fallzahlen in der frühen DDR als ›Tonnenideologie‹ markiert, die allein auf Stückzahl oder Masse, ohne Hinsicht auf Bedarf und Qualität abziele (vgl. Radkau 1990, 29). Diese war in der Frühphase der DDR durch den Marxismus-Leninismus der Sowjetunion vorgegeben und bestimmte für lange Zeit die Debatte, selbst als in der mittleren Leitungsebene Stimmen laut wurden, Produktivität anders als in Mengenangaben zu bestimmen. Sie waren gekoppelt an die Schaffung großer organisatorischer Strukturen wie die Volkseigenen Betriebe (VEB) oder Kombinate, die es erlauben würden, über die Betriebsebene hinaus Produktivkräfte miteinander zu kombinieren (Eppelmann u. a. (Hrsg.) 1997, Bd 1, Abschn Kombinat, 346f.). Diese Kombination von Produktivkräften sollte ab Mitte der 1970er Jahre allerdings auch eine größere Produktvielfalt und die häufigere Erneuerung der Produktstrecken ermöglichen, da sie die Koordination erleichtern und die Abhängigkeit von Zulieferern potentiell verringern sollte. Sie musste allerdings auf eine entsprechende Datenverarbeitungsinfrastruktur treffen, die Produktionsänderungen nicht nur abbilden, sondern auch simulieren und verändern konnte. An diesem Punkt reichten die Rationalisierungsbemühungen über eine reine Orientierung am Konzept ›Fließband‹ hinaus, da auch transaktionale Elemente eine Rolle

---

(ebd., 360). Zu Hübners Analyse ist kritisch anzumerken, dass sie grundsätzlich den Markt als Normativ setzt, und allein so das Argument entfaltet.

<sup>290</sup> Da in den SOPS-Beispielen nur Anwendungsbeispiele mit BASTEI-Nutzung und Prozessmanagementbezug erörtert wurden, konzentriert sich die Argumentation auf Management Resource Processing. Eine Diskussion von Informationssystemen wie dem SOPS AIDOS als Dokumentenverwaltungssystem in Firmen und Universitäts-Bibliotheken könnte zu anderen Schlussfolgerungen führen, bleibt vorerst Desiderat.

spielten, welche eine höhere Komplexität der Produktionsabläufe durch Einsatz von Informationstechnologien, wie beispielsweise den SOPS, ermöglichten. Spezifisch für die sozialistische DDR war, dass diese Vorgehensweise einer volkswirtschaftlichen und nicht allein betriebswirtschaftlichen Betrachtungsweise unterlag.

Zentralistisch wurde versucht, eine Standardisierung von Transaktionen im Sinne einer transaktionalen Rationalisierung zu steuern, wodurch topologisch ursprünglich voneinander getrennte Prozesse ineinander verfigt werden sollten. Die, zur Durchführung der Arbeitsprozesse, notwendigen Informationen bzw. Handlungsanweisungen sollten zum optimalen Zeitpunkt in immer gleicher Weise an den jeweiligen Arbeitsplatz oder Maschinenarbeitsplatz gelangen und die dafür notwendige ›händische‹ Informationsarbeit reduziert werden. Beispielhaft dafür steht die Standardisierung der Formulare, für die Kurt Schildhauer von der Zentralstelle für Statistik 1981 warb: »Der wesentliche Rationalisierungseffekt der standardisierten Primärdokumente liegt in der Vereinheitlichung von Begriffen, Stelligkeiten, Informationsstrukturen und der äußeren Gestaltung im volkswirtschaftlichen Maßstab [...] Mit der geringeren Vielfalt der bislang verwendeten und individuell gestalteten technologischen Vordrucke reduzierte sich die Verwaltungsarbeit. Das bezieht sich besonders auf das Erheben und Verdichten der unterschiedlichen technologischen Vordrucke für die Planung, die Bedarfsanforderung und die Beschaffung« (Schildhauer 1981, 5). Dies ist bereits an anderer Stelle ausführlich diskutiert worden, soll hier aber in Erinnerung gerufen werden, da es ein ganz wesentliches Element der Rationalisierungsmaßnahmen der DDR darstellte.

Viertens, ›informatische Alltagspraxen und kybernetische Operationsforschung‹: Neben dem, aus den Alltagspraxen heraus, marxistisch inspirierten Automatisierungsdiskurs gab es in der DDR auch einen kybernetisch hergeleiteten, der mit marxistischen Elementen synthetisiert wurde und sich auf das von Georg Klaus 1964 herausgegebene Monographie *Kybernetik und Gesellschaft* bezog. Betrieben wurde er vorrangig an der 1968 gegründeten Sektion Ökonomische Kybernetik und Operationsforschung (ÖKOF) der Humboldt-Universität Berlin in teils enger Zusammenarbeit mit dem medizinischen Zentrum der DDR, der Charité Berlin.<sup>291</sup> Der daran beteiligte Kybernetiker Klaus Fuchs-Kittowski kritisierte den frühen Automatisierungsbegriff der DDR als implizit geprägt durch die Idee der Vollautomatisierung. »In vielen Veröffentlichungen dieser Zeit wird die Automatisierung als höchste Form der Rationalisierung bestimmt« (Fuchs-Kittowski 2007, N. 37). Diese Kritik öffnete in den 1970ern den Weg für Überlegungen zur interaktiven Teilautomatisierung, wie hier kurz auszuführen ist. Im 1976 veröffentlichten Buch *Informatik und Automatisierung* griffen Fuchs-Kittowski et. al. die Automatisierungsdebatte auf und wiesen darauf hin, dass es nicht allein um die »systematische Anwendung der Naturwissenschaften« (Fuchs-Kittowski u. a. 1976, 1:24) ginge, sondern um neue Tätigkeitsfelder, die sich in der Auseinandersetzung mit maschineller Prozessierung eröffneten. Es stelle sich das Problem der »Konzipierung und gesellschaftlichen Realisierung neuer Tätigkeitsprofile und Organisationsstrukturen infolge der Automatisierungsmaßnahmen« (ebd.).

---

<sup>291</sup> Einen Einblick gibt der Protagonist Klaus Fuchs-Kittowski in (Fuchs-Kittowski 2007). Vielversprechend scheint die Auseinandersetzung mit der Monographie *Informatik und Automatisierung* (Fuchs-Kittowski u. a. 1976) zu sein, welche für die vorliegende Arbeit nicht ausreichend berücksichtigt werden konnte und als Desiderat seiner kritischen Neuaneignung durch die Medientheorie harrt.

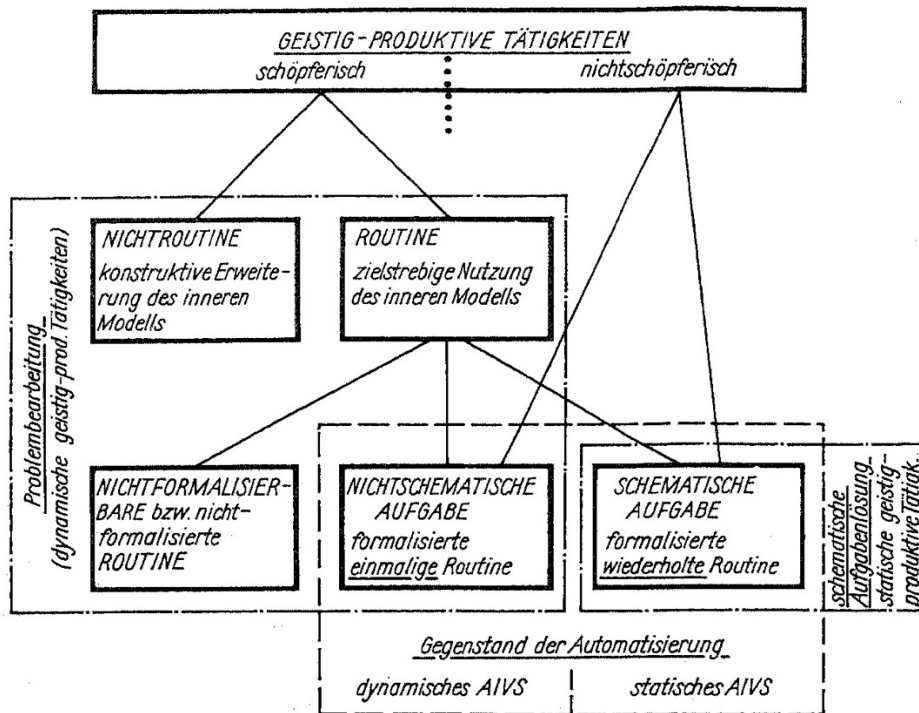


Abb. 66: »Gliederung der geistig-produktiven Tätigkeiten unter informationellem Aspekt« (Fuchs-Kittowski u. a. 1976, 1:248), Das Akronym AIVS steht für Automatisierte Informationsverarbeitungssysteme und ist den ASU und SOPS konzeptuell verwandt. Ursprünglich in (Fuchs-Kittowski/Tschirschwitz/Wenzlaff 1973, 118).

Auffallend an dem Diagramm (Abb. 66) ist, dass schöpferische und nicht-schöpferische Arbeitsanteile nicht streng voneinander getrennt, sondern als »formalisierte einmalige Routine« synthetisiert werden. »Der ständige, für die Automatisierung notwendige Umschlag von formalisierter Routine in automatisierte Abarbeitung kann nur durch den steuernden Eingriff des Menschen erfolgen, der in Abhängigkeit von seiner konkreten Problemstellung die erforderlich werdende Konkretisierung der Routine auslöst« (Fuchs-Kittowski u. a. 1976, 1:250, Hervorh. im Original). Dafür würde ein »dynamisches« Informationssystem benötigt.<sup>292</sup> Der damit aufgebaute Datenbestand wies, so Fuchs-Kittowski, in der Folge Potential auf, wenn Leitungserfahrungen mit neuen Methoden der »wissenschaftlichen Leitungstätigkeit« und der »Potenz der vielfältigsten Informationsverknüpfungen« (ebd., 407) kombiniert würden, womit neue organisatorische, aber auch mathematische Verfahren gemeint waren. Im Umkehrschluss wäre aber auch das Leitungsinformationssystem zu ergänzen und zu verbessern, da mit der Zeit erkannt werde, welche weiteren Informationen in die Datenbasis aufzunehmen sind. Der Rationalisierungsbegriff liegt hier also in einer Kombination der eigentlichen, sich verändernden Leitungstätigkeit mit neuen Informationsbedürfnissen, um so eine höhere Qualität der Leitung zu erreichen. »Diese Überlegungen sind gleichzeitig ein Ausgangspunkt für die Entwicklung von Modellen, die ganze

<sup>292</sup> Offen bleibt die Rezeptionsgeschichte von *Informatik und Automatisierung*. Hatte es über die akademischen Kreise hinaus Wirkungsmacht? Orientierten sich beispielsweise die Anwendungsentwickler bei Robotron an diesem konzeptionell anspruchsvollen und in ihrer Zeit höchst aktuellen Beiträgen? Eine mögliche Antwort liegt darin, dass die SOPS anfangs vor allem dem Manufacturing Resource Planning dienten, also der unmittelbaren Produktionsplanung, für die die Diskussion der Automatisierte Informationsverarbeitungssysteme für Leitungstätigkeit nur bedingt relevant war. Später wurden die SOPS um Module zur Leitungstätigkeit erweitert, z.B. durch die SOPS-Module Optimierung mittels Simplextechnik, Diskrete Optimierung, Transportoptimierung, Mathematische Statistik, Simulation von Systemen mit diskreten Ereignispunkten, Netzplantechnik oder Numerische Mathematik.

Systeme von Zusammenhängen zwischen Merkmalen repräsentieren und den konzipierten Datenbestand als Grundlage besitzen« (ebd., 407). Der Ort der Verknüpfung sei die Datenbank, auf die sowohl Leitungsebene also auch Anwender zugreifen könnten.

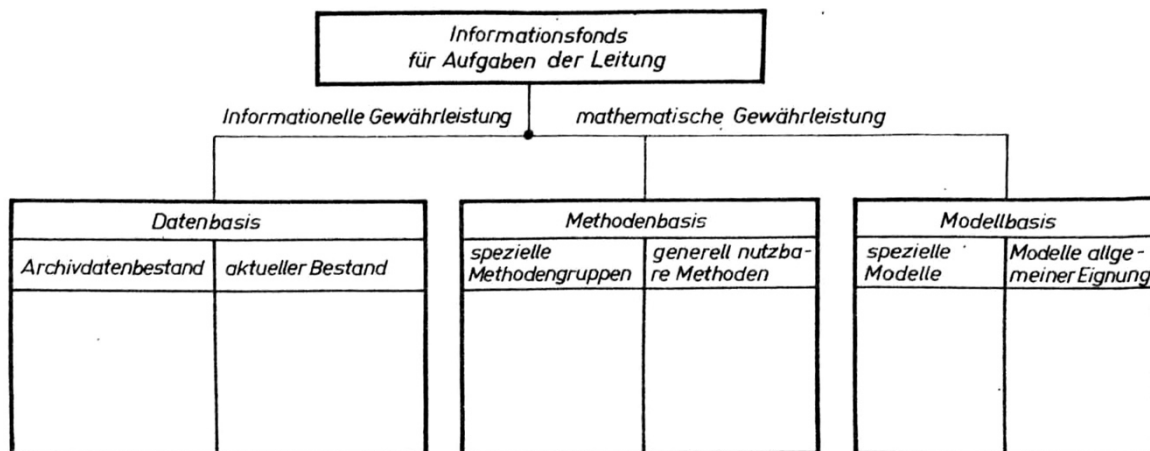


Abb. 67: Informationsfonds nach Christa Pflug. Die »Datenbasis« sollte softwareseitig durch die SOPS Bastei, SOPS DBS/R, SOPS AIDOS, SOPS SAWI abgesichert werden. Die »Methodenbasis« umfassten »Standardprozeduren für die Abarbeitung ökonomisch-mathematischer Aufgaben« und VOPS. Die »Modellbasis« enthielt »Erfassungs-, Erklärungs-, und Entscheidungsmodelle« anhand von »Nutzerkriterien« (Pflug 1976, 37, 39).

In diesen Überlegungen deutet sich das Bedürfnis nach veränderten Informationsflüssen für die Leitungsebene an, die zu Entscheidungssimulationen basierend auf Leitungsinformationssystemen, durch das »Bereitstellen von Kennziffern, Parametern, spezifischen Informationen, Weisungen« (Pflug 1976, 35), beitragen sollten. Dafür wird durch die Wissenschaftlerin Christa Pflug (TU Dresden) 1976 ein spezifischer Informationsbegriff aufgerufen: »Der Informationsfond für Aufgaben der Leitung stellt ein rechnergestütztes Hilfsmittel dar, das aus spezialisierten Daten- sowie Methoden- bzw. Modellbeständen besteht« (ebd.;

Abb. 67). Heute würde man statt von Methoden eher von Algorithmen sprechen. Demnach bestände der ideale Informationsfond zur Komplexitätsbeherrschung in Betrieben aus einem Repertoire an Daten, Algorithmen und Modellen, welche, so Pflug, Erzeugnisse und Leistungen betreffen. Den SOPS, also den Manufacturing Resource Planning-Paketen, stellt Pflug jene Elemente bei, die sich auf kybernetische Operationsforschung, Analyse und Simulation beziehen und in der DDR als Verfahrensorientierte Programmiersysteme (VOPS) und Verfahrensorientierte Programmpakete (VOPP) bezeichnet wurden. Diese überschneiden sich mit den SOPS, waren jedoch deutlich weniger bedeutend.<sup>293</sup> Wichtig ist dabei die Erkenntnis, dass es sich um »generell nutzbare Methoden« (Pflug 1976, 38) handelt, die also übergreifend, generalisierend und mithin im Sinne von Standards anwendbar sein sollten. Der Informationsfond sei, so Pflug, technologisch verwirklicht durch den Einsatz von Datenbanken (ebd., 39). Auffallend am Informationsfond ist, dass er der transaktionalen Ausformung des Rationalisierungsbegriffes zuarbeitet, denn er erlaubt sowohl die Verdichtung von Kommunikationsakten, als auch deren lokale Verteilung – auch wenn der Fonds selbst dann wieder zentral gedacht ist.

<sup>293</sup> Zu den SOPS gehörten nicht nur Manufacturing Resource Planning-Pakete, sondern auch die Datenbanken BASTEI, SAWI und DBS/R, die als gemeinsame Datenbasis für Abfragesysteme fungieren sollten. Weiterhin zählten dazu Module zur Planung und Optimierung, wie beispielsweise die SOPS Optimierung mittels Simplextechnik (OPSI), SOPS Diskrete Optimierung, SOPS Transportoptimierung, SOPS Simulation von Systemen mit diskreten Ereignispunkten (SIMDIS) oder SOPS Netzplantechnik (Autorenkollektiv/Gräßler/Freudenberg 1975).

Fünftens, »kybernetischer Durchgriff auf den Körper der Regierten«: Am Beispiel der ČSSR argumentierte der Ökonom und Mathematiker Miroslav Manas, warum Rationalisierungsschübe auf Leitungsebenen in der Planwirtschaft ins Leere laufen konnten. So wurde die Betriebsleitung unter dem Stichwort der Planübererfüllung dazu angehalten, die Produktion dauerhaft überschüssig zu fahren, ohne zusätzlich zu investieren. In den Studien zur Operationsforschung und in den Informatisierungsbemühungen ging es häufig darum, versteckte Reserven in Produktion und Kontrolle zu identifizieren. Diese stellten jedoch aus Sicht der Betriebsleitung einen dringend benötigten Puffer dar, um zwischen den staatlichen Planvorgaben, den tatsächlichen Produktionskapazitäten und den Interessen der Arbeiter in den Betrieben vermitteln zu können (Manas 1995, 432). Diese auf die DDR übertragbare Argumentation zeigt die Grenzen der Operationsforschung in der zentralistischen Planwirtschaft: Dort wo die Eigeninteressen des Leitungspersonals mit den Zielvorgaben kollidierten, schwächelt der gouvernementale Zugriff auf die Körper der Regierten. Die 1973 in *rechentechnik/datenverarbeitung* vom Sektorenleiter der Staatlichen Plankommission Günther Bretschneider geforderte »ungehemmte Nutzung dieser Ergebnisse der wissenschaftlich-technischen Revolution« (Bretschneider 1973, 5) traf ausgerechnet mit der untergeordneten Leitungsebene auf einen Hemmschuh, welche einheitliche Systemlösungen ausbremste, um eigene Spielräume zu erhalten.<sup>294</sup>

Sechstens, »transaktional Tiefenintegration mit post-fordistischen Produktionsweisen«: Kurz vor Ende der DDR, im Februar-Heft von *rechentechnik/datenverarbeitung* 1989 stand das Thema der Rationalisierung unter veränderten Vorzeichen zur Debatte. Die Grundlagen der Rechentechnik waren im Übergang von Groß- zu Kleincomputern und Personalcomputern begriffen und die Professoren der Sektion Sozialistische Betriebswirtschaft der TU Dresden, Siegfried von Känel, Hans-Walter Thiele und Werner Jöstel diskutierten »CIM« – Computer Integrated Manufacturing. Das ab Mitte der 1980er Jahre aufkommende Akronym CIM markiert in einer Parallelbewegung zu den Veränderungen der Rechentechnik, den Übergang vom MRP, dem Material Requirements Planning.<sup>295</sup> Die räumlich getrennte und über Formulare und Ausdrucke abgesicherte Informationsverarbeitung der Produktionsabläufe wurde durch die Verwendung von Bildschirmterminals und/oder Personalcomputern, die am Arbeitsplatz stehen oder den Maschinenarbeitsplatz unmittelbar robotisch integriert steuern, nach und nach abgelöst. Die Autoren konstatieren als Grund für die Einführung von CIM-Technologien die »Bedingungen der höheren Reproduktionsgeschwindigkeit« und das »Gewährleisten eines planmäßigen, bilanzierten und koordinierten Prozeßablaufs« (Känel/Thiele/Jöstel 1989, 15) und orientieren auf eine »Erhöhung *des Niveaus* der Produktion« (ebd., 16. Hervorh. F.H.).

Durch die Dezentralisierung von Client-Server-Systemen sehen sie, ähnlich den Entwicklern der SAP-Software in diesem Zeitraum, die Möglichkeit der »verteilten Datenspeicherung und Informationsverarbeitung« (ebd.). Ohne dass dies explizit durch die Autoren genannt wird, implizieren diese

---

<sup>294</sup> In den gesichteten Jahrgängen 1973–1989 der Zeitschrift *rechentechnik/datenverarbeitung* wird in den Editorials, aber auch in einer Vielzahl von Artikeln und Praxisberichten auf Nachnutzung Wert gelegt. Auch bei der Erstellung von Pflichtenheften und der Verteidigung fertiger Softwareprojekte vor Betriebs- und Plankommissionen musste die Frage der Nachnutzung dargestellt werden. Dies ist zumindest ein Hinweis darauf, dass es ein dauerhaftes Problem in der DDR war, die gewünschte Mehrfachnutzung von einmal programmierter Software auch tatsächlich zu erreichen.

<sup>295</sup> Während das Akronym MRP in der DDR-Literatur nicht erscheint, ist spätestens seit der offiziellen Absegnung des CAD/CAM-Programms durch den XI. Parteitag der SED im April 1986 die Verwendung englischer Akronyme Teil des informatorischen Diskurses der DDR. Computer Aided Design (CAD) und Computer Aided Manufacturing (CAM) sind Unterbegriffe des Computer Integrated Manufacturing (CIM). In der um Witze nicht verlegenen DDR wurde das Akronym CAD/CAM zu »Computer am Dienstag/Chaos am Mittwoch«.



Anforderungen die Verfügbarkeit von Datenbanken, welche transaktionssichere Zugriffsmöglichkeit für verschiedene, vernetzte Nutzerinnen bei gleichzeitiger Datenkonsistenz ermöglichen – eine Anforderung, die durch die aufkommenden relationalen Systeme, beispielsweise durch Robotron DABA 1600, MIMER, ALLDBS, Ing\_DB, Redabas und das in Entwicklung befindliche Robotron INTERBAS, erfüllt werden sollten.

Die Autoren identifizierten neue Anforderungen an die »Gestaltung der Planaufschlüsselung unter den Bedingungen einer rechnerintegrierten Produktion« (ebd., 16). Sie benannten neue Kriterien der Planung, die nicht länger allein an den Quantitäten des Produktionsausstoßes orientiert sind, sondern auch die Servicequalitäten der Informatisierung zuführen sollten: »Kennziffern zur Beurteilung der Qualität der Arbeitsergebnisse, zur Bewertung der Einhaltung von Terminen, von Normen des erzeugnis-, maschinen- und anlagenbezogenen Aufwandes« (ebd.). Zumindest für die DDR zeigt sich in dem Zitat eine Neuorientierung auf Qualitäten, welche in der Staatlichen Plankommission bisher eine eher untergeordnete Rolle spielten.

Der Artikel ist eine erstaunliche Überlagerung zweier Diskurse. Einerseits wird als erste Fußnote Karl Marx' Kapital angegeben<sup>296</sup> und als sprachliche Figur eine »noch wirksamere Einheit von Planung und wirtschaftlicher Rechnungsführung« (ebd.) beschworen.<sup>297</sup> Andererseits vermischen sich mit den realsozialistischen sprachlichen Phrasen Passagen, die in der DDR gegen Ende der 1980er Jahre erstaunlich klingen: »Reproduktion im Kontext zur *flexiblen, kundenorientierten* Fertigung von Erzeugnissen in relativ kleinen Produktionsmengen, [...] die Beachtung der wachsenden Bedeutung von *Qualifikation und Bildung*, der Arbeits- und Lebensbedingungen sowie *umweltreproduktiver Aspekte*« (ebd., Hervorh. F.H.). Damit dürften die aufgerufenen neuen Themen Flexibilität, Service und Umweltschutz, aktuelle Diskussionen in der Sektion Sozialistische Betriebswirtschaft an der TU Dresden darstellen. Nach eigener Aussage wollte sich die Sektion auf die operationale Planung einer rechnerintegrierten Produktion ausrichten und verstärkt in entsprechende Softwareprototypen investieren. Würde der DDR, mit Hilfe neuer Ansätze der Informationssysteme, der Datenbanken, der Vernetzung von Computern und interaktiver Zugriffsmöglichkeiten, der Übergang aus dem fordistischen Fließbandjahrhundert in eine globalisierte, post-fordistische Produktions- und Informationsgesellschaft gelingen?

Siebtens, »Individualisiertes Computing«: Während sich die Vereinheitlichung von Software mit den SOPS an Großrechnern zentral einigermaßen steuern ließ, mussten mit dem Auftreten der dezentralen Personalcomputer neue Strategien der Rationalisierung entwickelt werden. Für Kleincomputer, wie beispielsweise den K 1600, stellte Robotron bis 1984 analog zu den SOPS kleinere Varianten von Standardsoftware bereit. »Es wurden Lösungen für Datenorganisation, eine Modulbasis für Grundfunktionen, Software für Mathematische Verfahren und Betriebswirtschaftliche Komplexe (Lohn, Grundmittel, Kostenrechnung, Fakturierung) entwickelt« (Müller/Gräßler/Wildenhein In: Merkel/Junge/et. al. 2006, 114). Drei Jahre später, 1987, stand eine generelle Reorientierung der Sachgebietsorientierten Programmiersysteme an. In der Zeitschrift *Neue Technik im Büro* diskutierte Rolf Gräßler, der Leiter der SOPS, neue Beratungsleistungen, Dienstleistungen und Informationsangebote von Robotron, die

---

<sup>296</sup> Diese steht paradigmatisch für die naive, vereinfachende Marx-Rezeption in der DDR, so als habe Marx nicht eine Kritik der Politischen Ökonomie seiner Zeit entworfen, sondern eine Vorlage für eine sozialistische Betriebswirtschaft.

<sup>297</sup> Der Komparativ »noch« ist hier entscheidend: »die Steigerungsfigur soll ja Fortschritt anzeigen, und zwar als eine lineare Entwicklung« (Cosentino/Ertl/Labrousse (Hrsg.) 1988, 26:131). Der inflationäre Gebrauch dieser diskursiven Figur in der DDR führte jedoch dazu, dass sich die Steigerungsform »so verselbständigt hat, daß sie letztlich auch ihre Unterscheidungskraft verliert [...]« (ebd. 133).

einen neuen Abschnitt der Computernutzung – den Einsatz direkt am jeweiligen Arbeitsplatz – begleiten sollten.

Er bezog sich auf die Lieferung des »arbeitsplatzorientierten« PC 1715, seit 1986 und plädierte auch hier dafür, »nicht erst den zeitaufwendigen Weg der Eigenprojektierung« (Gräßler 1987, 76) zu gehen, sondern vorhandene Software nachzunutzen.<sup>298</sup> Sicherergestellt sollte dies dadurch werden, dass eine zentrale Informationsdatenbank für Software beim VE Kombinat Datenverarbeitung Dresden, sowie eine Sachgebietsorientierte Informations- und Beratungseinrichtungen beim VEB Robotron Projekt eingerichtet werden sollten.<sup>299</sup> Laut Gesetz bestand eine Pflicht zur Meldung von Eigenentwicklungen. »Der Softwareentwickler erhält damit die Chance, ausgehend von der Meldung seiner Software an die Zentrale Informationsbank Software, diese Erzeugnisse für den Export einzubringen« (ebd., 77).

In Reaktion auf die Veränderungen der Rechentechnik ging die DDR neue Wege: Die eingerichteten Stellen sollten nicht nur der Informationssammlung, sondern auch der Beratung für potentielle Nachnutzer dienen. Damit löste man sich von der industriellen Produktion von Software im Rahmen des Kombinats Robotron und erkannte die zunehmende Bedeutung der einzelnen User an. In diesem Sinne diente die Zentrale Informationsbank Software (ZIS) der Reintegration der, aus den Rechenzentren an die Arbeitsplätze entstrebenden, »unmittelbaren Produktivkraft Wissen« in die Bilanz der Planwirtschaft.

Verbunden mit der Aufforderung zur zentralen Erfassung war das neue Angebot von Schulungsveranstaltungen für »Basissoftware« wie die Textverarbeitung Text 30 (Adaption von Text Processor/TP), das Datenbanksystem Redabas (Adaption von dBase), Tabellenkalkulation mit KP/TABCALC und das Informationsrecherchesystem AIDOS, eine Eigenentwicklung, welche nicht nur für Groß- und Kleinrechner, sondern auch für DDR-PCs zur Verfügung stand (ebd.). Tatsächlich ist in den Ausgaben der Fachzeitschrift *rechentechnik/datenverarbeitung* eine Zunahme von Kurzmeldungen und auch Kleinanzeigen-Gesuchen über nachnutzbare Software zu verzeichnen, die Folge zunehmender PC-Nutzung und zum Teil auch der Anordnung zur Nachnutzung gewesen sein dürften.

Nach dem Durchgang durch diese Aspekte des DDR-Rationalisierungsbegriffes können mehrere Gemeinsamkeiten und Unterscheidungen herausgearbeitet werden. Der traditionelle Rationalisierungsbegriff ist stark orientiert an der Logik der Fließbandproduktion, aufbauend auf Adam Smiths Beobachtung der Arbeitsteilung,<sup>300</sup> auf einer Erhöhung der Taktzahlen und Reduzierung unproduktiver Zeiten, wie zum Beispiel der Produktvorbereitung, der Werkzeugwartung, dem Warten auf Materialien, die noch geliefert oder im gleichen Betrieb hergestellt werden müssen, das »Ideal der Fließarbeit« (Vahrenkamp 2017, 235).

Für die DDR war dieses Rationalisierungskonzept genuin sozialistisch gerahmt durch das Marx'sche Konzept der »Produktivkraft Wissenschaft«. Die Rationalisierung trat für die DDR-Volkswirtschaft vorrangig als Krisentaktik auf: Da in der DDR nicht die Einzelkapitalien gegeneinander antraten, zielte

---

<sup>298</sup> Die Nachnutzung von Software war ein dauerhafter Diskurs und wurde sowohl gesetzlich in der *Anordnung über die Nachnutzung wissenschaftlich-technischer Ergebnisse innerhalb der Deutschen Demokratischen Republik* (Der Staatssekretär für Forschung und Bildung 1967), als auch in zahlreichen Artikeln in *rechentechnik/datenverarbeitung* thematisiert, zum Beispiel in: *Was sind SOPS, Typenprojekte und Software?* (Sack 1974), *Vorhandenes wirksamer nutzen* (Loll 1982), *Softwarenachnutzung – Ziele und Mittel* (Claus 1987).

<sup>299</sup> Abgesichert war diese Strategie durch die *Anordnung über Informations- und Beratungsleistungen zur Entwicklung, Produktion und Mehrfachnutzung von Software in der DDR* (Der Minister für Wissenschaft und Technik/Der Minister für Elektrotechnik und Elektronik/Der Leiter der Staatlichen Zentralverwaltung für Statistik 1986).

<sup>300</sup> Vgl. *An Inquiry into the Nature and Causes of the The Wealth of Nations* (Smith 1776, 1–33).

Rationalisierung vor allem auf die Abwendung der wirtschaftlichen Krisen der gesamten Volkswirtschaft. Das Rationalisierungskonzept war auf die Skalierung der gesamten industriellen Produktion gerichtet, ein Vorhaben, welches mit der ungenügenden Übereinstimmung von Konsumbedürfnissen im Zuge der ›Tonnenideologie‹ an seine Grenzen geriet. Die anfangs stark vertretene Idee der Vollautomatisierung wurde mit der Zeit ersetzt durch eine Debatte um neue integrierte Arbeitsansätze. Dazu zählt das Ineinandergreifen kybernetischer Operational-Research-Ansätze mit Alltagspraxen der Informationsverarbeitung, wie von Fuchs-Kittowski und Pflug herausgearbeitet wurde. Retrospektiv lassen sich jedoch auch hier Grenzen erkennen, die in der aktuellen Socialist Calculation Debate (vgl. Morozov 2019b) Beachtung finden sollten: Das mittlere Management war um die Erhaltung von Spielräumen bemüht und machte daher ›falsche‹ Angaben – ein vollständiger gouvernementaler Zugriff war, so die Erfahrung, in der Planwirtschaft nicht möglich. In den späten 1980er Jahren wurde in der DDR, die über die Teilnahme am Weltmarkt fortwährend in die Konkurrenzverhältnisse des Kapitalismus eingewoben war, eine zunehmende transaktionale Tiefenintegration mit post-fordistischen Produktionsweisen angestrebt. Dazu zählte die Übernahme von Server-Client Architekturen und die Verbreitung von Personalcomputern, welche neue Transaktionsbedürfnisse schufen und ermöglichten. Mit Software-Verzeichnissen zur Nachnutzung wurde versucht, den sich verändernden Softwarefigurationen aus Rationalisierungssicht entgegenzukommen.

Neben den Unterschieden im Rationalisierungskonzept, die aus der sozialistischen Produktionsweise resultierten, gab es zahlreiche Überschneidungen mit dem ›kapitalistischen‹ Konzepten von Rationalisierung, da beide Systeme stark auf Industrieproduktion setzten. Daher lässt sich das Konzept von Rationalisierung mit Blick auf die zunehmende Transaktionalisierung für beide politische Systeme aktualisieren. Rationalisierung tritt dabei in zwei Phasen auf:

- erstens, der bisher bereits breit diskutierten fließbandorientierten Rationalisierung und
- zweitens, einer transaktionsorientierten Rationalisierung.

Während erstere auf die Verdichtung, Zerlegung und Standardisierung der Arbeitsabläufe abzielte, orientiert sich zweitere stärker auf das Delegieren mit Hilfe von Informationen. Diese Veränderung korrespondiert mit dem Aufkommen post-fordistischer Elemente der Informations- und Dienstleistungsgesellschaft. Koordinierende Informationsströme sowohl in der Warenproduktion, als auch in der entstehenden Dienstleistungsgesellschaft reorientierten das Rationalisierungsverständnis entlang der Transaktion, welche nicht allein die Abfolge der Ereignisse, sondern auch deren vollständige Abwicklung erst formatisierte und dann informatisierte, um dem komplexen Koordinieren von lokal verteilten Prozessanteilen Genüge zu tun. In der Folge sind, mit dem Überschreiten des Zenits der Mechanisierung, die frühen Ausläufer einer globalisierten Wissensgesellschaft zu beobachten. Wichtig an einem transaktionalen Rationalisierungsbegriff ist, dass Rationalisierung einerseits als Verfügung verteilter Kommunikationsakte und andererseits als horizontale Vernetzung von Akteuren gefasst wird.

### 3.2.7 Fazit

Erstens, ›Konzeptuelle Kontinuitäten‹: Mit den SOPS verfügte die DDR in den 1970er Jahren über ein innovatives, modulares Softwaresystem für Großrechner. Die SOPS kamen in der Industrie zum Einsatz und stehen für eine Reihe von Systemen, in welche bürokratische, manageriale und koordinierende Praktiken eingeschrieben waren. Die Argumentation, dass heutige ERP-Figurationen im Kern immer noch Funktionalitäten der frühen Vorfahren aufweisen und daher eine konzeptuelle

Kontinuität zu konstatieren sei (vgl. Pollock/Neil/Williams 2009, 25), ist hier durch zahlreiches Material belegt. Daher ist die intensive Diskussion von SOPS, COPICS und SAP R1/R2 als genealogischer Vektor angelegt, der vom frühen Material Requirements Planning zu heutigen ERP, SCM und CRM-Softwares führt.

Keineswegs waren die SOPS eine monolithische Software, im Sinne eines einzelnen Programmes, welches alle Aufgaben erledigte. Vielmehr bestand jedes SOPS aus vielen Einzelprogrammen, die untereinander mittels definierter Schnittstellen interagierten. Diese Datenbank-ERP-Figurationen stellen daher ein Beziehungsgeflecht der Teilelemente dar. In jedem der Teilelemente waren wiederum Objekt-Prozess-Beziehungen eingeschrieben. Die Objekt-Prozess-Beziehungen bildeten Produktionsprozesse ab. Datenbank-ERP-Figurationen beschreiben daher ineinander verflochtene Beziehungen von Beziehungen, welche beständig aktualisiert werden. Erst diese Bezüglichkeit, die Relationalität, macht sie zur idealen technologischen Grundlage für die Globalisierung und Dynamisierung von Geschäftsprozessen, die in ihrer Komplexität mit manuellen, nicht-automatisierten Mitteln nicht darstellbar wären. Der ›östliche‹ SOPS-Einsatz in den Ländern des Einheitlichen Systems Elektronischer Rechentechnik (ESER) zog die Notwendigkeit der Adoption lokaler Gegebenheiten, z.B. Sprache und Währung nach sich und auch für die ›westliche‹ SAP lässt sich nachweisen, wie die Einführung bei international tätigen Firmen das Anforderungsprofil an die Softwaremodule änderte und gleichzeitig die Firmen in der Abwicklung ihrer globalen Tätigkeit unterstützte.

Die Foki der Softwareentwicklung waren verschieden gelegt: Geht es für die SAP im Westen darum, Geschäftsprozesse in Echtzeit zu integrieren, also das Online-Potential von Computern auszuschöpfen, so lag die Motivation für die SOPS im Osten vorrangig in der betriebsinternen Rationalisierung und der Mehrfachverwendung gleicher Module. Beiden ist gemein, dass sie über das Fließband hinaus erlaubten, Produktionsprozesse transaktional zu organisieren, sodass eine Diversifizierung und Globalisierung überhaupt ermöglicht wurde. Das Potential der Globalisierung von Warenströmen schöpfte die DDR nicht aus.

Zweitens, ›Datenbanken als zentrale Datenreservoir‹: Gemein ist den untersuchten modularen Systemen, dass sie Datenbanken als zentrale Datenreservoir benötigten und hervorgebracht haben. Diese war wichtige infrastrukturelle Voraussetzung für die angestrebte Skalierbarkeit der SOPS, bzw. genereller von ERP-Figurationen. Die Softwareinfrastruktur zeichnete sich durch eine hohe Dynamik aus: Mit der engen Abfolge von Gerätegenerationen in den 1960er bis 2000er Jahren ging eine beständige Überprüfung und Erneuerung der Software einher, die zu einer Periode intensiver infrastruktureller Updates führte. Damit blieben die Ordnungen des Wissens in beständiger Schwebelage und bildeten Schichtungen und Überlagerungen in den ERP-Figurationen und den Datenbasen. Hellsichtig beschrieb Joseph Orlicky die Datenhaltung innerhalb von Datenbanken als automatisiertes In-Ordnung-Halten (Orlicky 1974, 190) und betonte damit noch einmal die Automatisierung der Datenordnung und -verortung zwischen Stabilisierung, Fixation und Aktualisierung. Wichtig ist jedoch: Die Einsicht in den Nutzen standardisierter Datenreservoirs setzte sich nur langsam durch. Bei Robotron beispielsweise bezogen sich die verschiedenen SOPS nicht durchgehend auf die Datenbanken DBS/R oder BASTEIL. Abhängig war die Kooperation in dieser Frage von den Abteilungsleitern der verschiedenen SOPS. Die Konzeption der Datenbanken als zentrales Reservoir war in den SOPS nicht von vornherein klar, sondern schälte sich erst im Gebrauch über einen langen Zeitraum heraus.

Drittens, ›Standardisierung‹: Eine weitere Voraussetzung von Skalierbarkeit war die Standardisierung, die es erlauben sollte, einmal produzierten Code zu kombinieren und in verschiedenen

Anwendungszusammenhängen zu platzieren. Die beschriebenen Momente des Übergangs, z.B. der Einführung der SOPS in den Betrieben, lassen die Affordances und die Materialitäten von frühen Datenbank-Infrastrukturen sichtbar werden.

Die Formulare der einheitlichen Primärdokumente weisen mediale Kontinuitäten der Sammlung und Verarbeitung von Information in Tabellen auf, die eine tiefenhistorische mediale Qualität annehmen und die – nach entsprechender organisatorischer Anpassung der betrieblichen Realität – in die Datenstrukturen der ERP-Figurationen übergangen. Standardisierung und Formalisierung im Vorfeld von Algorithmisierung, Modellierung und Datenhaltung treten als mediale Materialien auf, denen weitere medientheoretische Aufmerksamkeit gebührt.

Viertens, ›Informationsmodell und Materialität von Daten‹: Mit Pflug (1976) und Fuchs-Kittowski (1976) lässt sich argumentieren, dass die Abbildung organisatorischer Prozesse ein formalisiertes Modell voraussetzt, welches sowohl Zustände als auch Dynamiken abbilden kann. In den ERP-Figurationen stehen nicht Algorithmen im Vordergrund, sondern es bedarf des Zusammenspiels von Daten, Modell und Algorithmen. Pflugs und Fuchs-Kittowskis Beiträge bilden insofern Ausgangspunkte für ihre weitere medientheoretische Wiederentdeckung.

Daten bespielen in den Fallbeispielen unterschiedliche topologische Register: Stammdaten beschreiben Objekte oder Subjekte und sind, einmal eingegeben, relativ wenig veränderlich. Prozessuale, transaktionale Daten hingegen unterliegen häufigen Aktualisierungen. Sie altern oder werden gelöscht und ersetzt. Häufig beschreiben sie eine zeitliche-räumliche Dimension und bilden menschliche Handlungen ab. Analysedaten verknüpfen die topologischen Informationen der Stammdaten mit den prozessualen Informationen der Transaktionsdaten. Die Verknüpfung aus zeitlichen und sachlichen Informationen schafft in Analysedaten neues, spezifisches Wissen. Mit diesen Unterscheidungen sind unterschiedliche Materialitäten von Daten aufgerufen.

Fünftens, ›Reorganisationen der Ordnungen des Wissens‹: Mit Schüttpelz/Gießmann lässt sich die organisationstheoretische Frage als Praxen des Registrierens, des Delegierens, und der Koordination fassen und die daraus resultierenden Medien als Medien der Kooperation. In diesen Praxen spielt die Formalisierung als Information eine medienbestimmende Rolle. Mormann beschreibt einen doppelten Begriff der Formalisierung: zum einen als informationstheoretisch begründet, als Formierung von Informationen, die mathematisch verarbeitbar werden und zum anderen organisationstheoretisch als Regulation von Kommunikationsabläufen durch Hierarchien und Arbeitsabläufe (Mormann 2013, 70). Dieser doppelte Formalisierungsbegriff ist für SOPS, MRP und ERP produktiv, adressiert er doch die Verknüpfungen von Zeitlogiken (in Bezug auf Information) und topologischen Ordnungen (in Bezug auf Organisation) in Datenbanken, die bereits im Abschnitt über die doppelte Buchhaltung eine wesentliche Rolle spielten.

Die damit verbundenen, gezielt funktionalisierten Reorganisationen der Ordnungen des Wissens laufen auf eine Dynamisierung des Wissens hinaus, welches sich als ständig aktualisiertes, gleichwohl formalisiertes Prozesswissen neu konstituiert. In einer Nutzungsperspektive fällt auf, dass Manager und Programmierer neue Wissensfelder generieren, welche organisatorische und technologische Beziehungen betreffen. Das Nutzungswissen der Anwender wiederum ermöglicht es, die im Modell eingeschriebenen Formalisierungen von Realität in die vorgefundene materielle und operationale Situation zu vermitteln, wobei es gerade diese Vermittlungsfunktion ist, welche die Aufrechterhaltung der Softwareinfrastruktur überhaupt erst erlaubt.

Sechstens, ›(Sozialistischer) Automatisierungsdiskurs der DDR‹: In der DDR waren Automatisierung und Nachnutzung vorhandener Software feste Bestandteile des offiziellen Rationalisierungsdiskurses, der der Ressourcenschonung dienen sollte. Der ›Arbeiter- und Bauernstaat‹ nahm dabei für sich in Anspruch, rationalisierte Arbeitskraft zunehmend ›schöpferischen Tätigkeiten‹ zuführen zu wollen – von Arbeitslosigkeit sei niemand bedroht. Begrenzt wurden die Rationalisierungsinitiativen durch Akteure des mittleren Managements, welche, statt einer weiteren Intensivierung von Prozessen, Spielräume im Feld der Planwirtschaft aufrechterhalten wollten. In der Konzeption der SOPS wurde die These vertreten, dass der Einsatz westlicher Software nicht schaden könne, da sie in einem völlig anderen Nutzungskontext, einem sozialistischen, auftrete. Unausgesprochen blieb dabei, dass Nutzungen – im Kapitalismus, und auch im Sozialismus – durch Industrieproduktion gekennzeichnet waren, die sich im Übergang zu post-industriellen Produktionsweisen befand.<sup>301</sup> Die Robotron-Ingenieure waren aus zweierlei Richtung getrieben: Einerseits durch die Kader in den Staats- und Parteiapparaten und deren Forderungen nach allgemeiner Rationalisierung und andererseits durch ihre eigene Kenntnis der Entwicklungen im ›Westen‹, welche zu einer Situation des permanenten Selbstvergleiches führte. Darin unterschieden sich die Firmenkulturen von IBM und Robotron. IBM besaß bis Ende der 1970er Jahre Marktdominanz und musste dann vor der ›jüngeren‹ Konkurrenz weichen. Nicht zufällig fällt dieser Wandel IBMs mit dem Abflauen des Industriezeitalters zusammen. Für Robotron bestand in einem Land, welches lange an der Industrialisierung als Konzept festhielt, dieses Problem nicht unmittelbar. Vielmehr erschien hier der Abschwung der Industrialisierung mittelbar, durch den Vergleich mit dem Westen qua Teilnahme der DDR am internationalen Marktgeschehen im Zuge von Exporten.

Siebtens, ›Transaktionale Rationalisierung‹: Im Zuge der vorliegenden Untersuchung erwies sich ein am Fließband orientierter Rationalisierungsbegriff als nicht ausreichend. Es zeigte sich, wie dieser erweitert und aktualisiert werden kann. Für den untersuchten Zeitraum ab den 1960er Jahren ist ein transaktionsorientierter Rationalisierungsbegriff, der die delegierenden und koordinierenden Agencies von Datenbankfigurationen beinhaltet, besser geeignet, die zeitlich orientierten und lokal verteilten ineinandergreifenden Prozesse zu beschreiben. Maßgeblich ist dabei der Umstand, dass eine Transaktion immer vollständig sein muss und gegebenenfalls auch rückabgewickelt werden kann. Diese ineinandergreifende Reversibilität ermöglicht komplexere Automatisierungen und Rationalisierungsformen, als die fließbandorientierte Abwicklung eines Vorganges.

Achtens, ›Daten-Positivismus in Ost und West‹: Die in den Figurationen eingeschriebenen Formalisierungen von Realität lassen die abgebildeten Prozesse als planbar erscheinen und werden dadurch zum Normativ, an dem sich die betrieblichen Abläufe orientieren sollen. Es ist eine Objektivierung von datengetriebenen Einschreibungen beobachtbar, wobei Daten als objektives Maß gesetzt werden und eine positivistische Normativität entlang der erhobenen Daten erzeugen. Diese Normativität konnte sich auch kontraproduktiv auswirken, denn die Planwirtschaft der DDR war ohnehin geplagt davon, dass die Planzahlen dekontextualisiert abgearbeitet wurden – teilweise wider besseren Wissens und so an

---

<sup>301</sup> Die Verwendung des Präfixes »post« in »post-industriell« ist problematisch. Es könnte suggerieren, dass hier eine Abfolge gemeint ist, dernach der industriellen Produktion die Wissensgesellschaft folgt, und dass eine Formation die andere ablöst. Gemeint ist vielmehr, dass der Focus auf industrielle, fordistische Produktionsweisen in unterschiedlichen geographischen Verteilungen partiell schwindet und parallel dazu jene Formationen im Entstehen sind, die als globalisierte »Wissensgesellschaft« (Lane 1966; Bell 1974) charakterisiert werden. »Post-industriell« bezeichnet in der Verwendung hier den Umstand, dass industrielle Produktionsweisen nicht mehr allein oder hauptsächlich die ökonomische Sphäre dominieren. Industrielle Produktion wird lediglich geographisch anders verteilt (vgl. Soja 2000, 164f.).

den aktuellen volkswirtschaftlichen Notwendigkeiten vorbei. Für den vom Marktgeschehen dominierten Westen hingegen bedeutete diese positivistische Normativität eine beständige Fortschreibung und unter Konkurrenzverhältnissen, Verschärfung, ökonomischer Verobjektivierung gesellschaftlicher Beziehungen.

Der Daten und Informationspositivismus mündete in einen Technopositivismus und so konstatierte über die 1950er und 1960er Jahre der Historiker Thomas Haigh für die USA: »Der weit verbreitete Glaube an die Praktikabilität und den Nutzen eines vollständig integrierten Management-Informationssystems [...] wurde gekennzeichnet durch den Glauben an die Vorteile von Größe, zentraler Planung, Rationalität und Technologie, um jegliche Probleme zu lösen« (Haigh 2001, 57).<sup>clxviii</sup> Ähnliches kann für sozialistische Planungsphantasien konstatiert werden. Ende der 1970er Jahre wurden diese zwar einer internen Kritik und Revision unterzogen, wie im Abschnitt zur Rationalisierung sichtbar gemacht, doch durchdrangen Planungsphantasien den Diskurs bis zum Ende der DDR in großem Maße.

Prekär blieb die unmittelbare Aneignung von Konzepten und Software durch Kopie oder Adaption aus dem Westen. Der Historiker Simon Donig konstatiert, dass dies nur möglich war, »weil auf beiden Seiten des Eisernen Vorhangs grundlegende Vorstellungen wie Effizienz, Kosten-Nutzen-Relationen oder Rationalität geteilt wurden« (Donig 2009, 100). Die ideologischen Vorgaben, welche den DDR-Sozialismus dem BRD-Kapitalismus überlegen sehen wollten, setzten diese Praxen unter Spannung – eine Spannung die produktiv und hemmend zugleich war.

### 3.3 Kalifornien, Technische Universität Dresden und Robotron Dresden: Das Problemorientierte Seminar »Datenbanken« und DABA 1600

Die Genese des relationalen Datenbanksystems DABA 1600 nahm an vielen Orten gleichzeitig ihren Anfang: Sie begann im Ministerrat der DDR und im Zentralkomitee der SED mit einer staatlichen Software-Konzeption, welche eingebunden war in Vereinbarungen des RGW und abgestimmt mit den Betriebsdirektoren (Merkel 2005, 55).

Sie begann 1970 in den über 9000 km entfernten, durch den Eisernen Vorhang und das COCOM-Embargo abgeschnittenen Forschungszentren der IBM in San Jose. Nach den Veröffentlichungen zur relationalen Algebra arbeitete es an einem Anwendungsprototypen System R. Sie begann an den Schreibtischen, an denen die Forscher Veröffentlichungen zu System R verfassten, in den Redaktionsräumen der Verlage, welche die Paper zur Publikationsreife brachten und in den Druckerpressen, welche die Ideen auf Papier vervielfältigten. Bibliothekare der DDR-Staatsbibliothek Unter den Linden, Berlin, der Universitätsbibliothek Dresden und der Arbeitsbibliothek des VEB Robotron abonnierten Publikationen wie zum Beispiel die *Communications of the ACM* und stellten sie in den Lesesälen als Quellen zur Verfügung. Dazu gehörten auch Veröffentlichungen westdeutscher Akademiker und Praktiker, die ihr, in Kalifornien vor Ort erworbenes Wissen über das System R, in deutschsprachige Publikationen in der Bundesrepublik fließen ließen, welche wiederum ihren Weg in die DDR fanden.

Sie nahm ihren Ausgang in Seminaren, Dissertationen und Diplomarbeiten an der Fakultät Elektrotechnik/Elektronik der Technischen Universität Dresden und mit Betriebspraktika der Studierenden beim ZFT Robotron.

Die Genese von DABA 1600 begann aber auch mit dem Verständnis der Verantwortlichen, dass für den bei Robotron in Entwicklung befindlichen Rechner K 1600 entsprechende Software benötigt werden würde. Es handelte sich beim Robotron K 1600 um einen sogenannten Kleinrechner, vergleichbar der Leistungsklasse der ›westlichen‹ DEC PDP-11. Er basierte auf dem in der DDR eigens entwickelten und in großer Stückzahl produzierten 8-Bit Chipsatz U-830 (Merkel 2005, 32). Diese ostdeutsche Nacherfindung war kompatibel mit dem ›westlichen‹ Zilog Z80 Chip. Die Weiterentwicklung und Ausreizung dieses Chipsatzes, dessen Variante erstmals 1978 hergestellt worden war, bedeutete gegenüber der zuvor eingesetzten Transistortechnik: Miniaturisierung.

Um ihn in Betrieb zu nehmen, kam auf dem K 1600 das Betriebssystem MOOS 1600 (Modular Operating System) zum Einsatz, welches durch den U-830 Chip mit Einschränkungen kompatibel zum Betriebssystem des PDP-11 Computer war. Aus dem Umstand, dass keine vollständige Kompatibilität erreicht wurde, entsprang der Bedarf an einem eigenen Datenbankbetriebssystem, denn ›westliche‹ Produkte konnten nicht ohne weiteres verfügbar gemacht werden, wie dies auf anderen vollkompatiblen Computern der DDR der Fall war.<sup>302</sup> »DABA 1600 wurde für das MRS Robotron K 1630 mit einem Hauptspeicher von 256 kByte entwickelt. Zur Minimalkonfiguration gehören ein Bedienterminal K 8911, zwei Kassettenplattenspeicher CM 5400 und eine Magnetbandeinheit CM 5300. Außerdem unterstützt DABA 1600 weitere Direktzugriffsspeicher, die Terminals vom Typ K 8912 und

---

<sup>302</sup> Erst einige Jahre nach dem Betriebssystem MOOS 1600, wurde MUTOS 1360 (MultiUser Time Sharing Operating System) in der DDR veröffentlicht. Diese Adaption des ›westlichen‹ UNIX-V brachte Mehrnutzerfähigkeiten mit sich und konnte potentiell auf einer ganzen Reihe von DDR-Rechnern eingesetzt werden. Allerdings schätzen die Autoren von robotrontechnik.de ein, dass MUTOS insgesamt wenig verbreitet war. In der Konfiguration mit MUTOS hätte auch die Verwendung des in Berkeley entwickelten Datenbanksystems INGRES möglich sein können. Mit ING\_DB, INDES und DABA 32 standen ab 1986 in der DDR INGRES-Clone zur Verfügung.



Bürocomputer. Als Betriebssystem wird das Modulare Operationssystem 1600 (MOOS 1600) verwendet. [...] Es ist auch möglich, DABA 1600 auf Rechnern wie CM 4, CM 52, I 100, I 102 und PDP-11 zu verwenden [...]« (Menhart/Stumpf 1987a, 18).



Abb. 68: K 1600 Computer in einer sowjetischen Forschungseinrichtung, Mitte der 1980er Jahre in Kirow (Alexander Sacharnych, o.J., privat)

Der Rechner K 1600 (Abb. 68) fand in unterschiedlichsten Konfigurationen in der DDR und später auch im RGW Verbreitung und wurde für Alltagsaufgaben eingesetzt, zum Beispiel für Büro- und Verwaltungsaufgaben, für die Textverarbeitung, Konstruktions- und Digitalisierungsaufgaben, sowie für Datenerfassung in Rechenzentren, Handel und Medizin (Merkel 2005, 31).

Im Unterschied zu Adaptierungen, also Aneignungen, Modifizierung und Neupublikation ›westlichen‹ Quellcodes, erlaubt eine Darstellung von DABA 1600 als eigenständige Entwicklung, Prozesse des Wissenstransfers über geographische und politische Systemgrenzen hinweg zu erörtern. Auch wenn der Eiserner Vorhang (1948–1989) spezifische Bedingungen hervorrief, kann dieser Wissenstransfer als paradigmatisch für Softwareverwendung und -herstellung in kleineren Ländern mit beschränkten Ressourcen gelesen werden. Er ist kennzeichnend für die verschiedenen Modi und Abstufungen von Eigenständigkeit in technologischen Entwicklungen: Phasen von Transfer, Kooperation und Publikation wechseln sich ab mit Momenten individueller Problembewältigung und institutioneller Abgrenzung. DABA 1600 stellt ein großartiges Beispiel dafür dar, den kompletten Lebenszyklus eines Datenbanksystems von den Vorarbeiten, über die konkrete Ausarbeitung bis hin zur Lieferung an die Endverbraucher, unter den Bedingungen der sozialistischen Planwirtschaft zu erklären.

### 3.3.1 Ein Informatik-Zentrum der TU Dresden

An der TU Dresden begründete vor allem die dritte DDR-Hochschulreform von 1968 maßgeblich eine Strukturänderung und die Einführung der Informatik in die Lehre.<sup>303</sup> Vorarbeiten für einen Studiengang ›Elektronische Datenverarbeitung‹ an der Fakultät für Elektrotechnik und Informationstechnik begannen ab 1968 durch das Ministerium für Hoch- und Fachschulwesen und eine Arbeitsgruppe Datenverarbeitung der Fakultät. In ihr waren sowohl Lehrende als auch Industrievertreter präsent. Man forderte eine Umprofilierung der Elektrotechnik in Elektronische Datenverarbeitung (Pulla 2003, 248). Daraus folgte die Gründung der Sektion 08 Informationsverarbeitung mit neu zu besetzenden Professuren und einer auf vier Jahre verkürzten Studienzzeit. Den damit verbundene Abschluss als ›Hochschullingenieur‹ sah man als Berufsbefähigung vor. Häufig schlossen die Studierenden noch eine einsemestrige Diplomarbeit und ggf. ein Forschungsstudium oder eine Aspirantur an, die zur Dissertation führte (Schmidt 2008, 148).

Begründet wurde die Notwendigkeit einer neuen Sektion Informationsverarbeitung in der DDR mit der Erkenntnis, dass nicht allein Hardware, sondern die damit zu realisierenden Informationssysteme wesentlich für Produktions- und Rationalisierungsvorhaben waren. Die Sektion sollte zudem nicht nur der Grundlagen- und Anwendungsforschung dienen, sondern auch in die anderen Sektionen hineinreichen und deren Informatik-Grundausbildung absichern (Sonnemann (Hrsg.) 1978, 339). Sie wurde in drei Bereiche untergliedert: 1.) Automatisierte Informationssysteme, wozu auch das Forschungsfeld Datenbanken gehörte, 2.) Programmierungstechnik und 3.) Rechnersysteme. Die Räumlichkeiten befanden sich in der Dürerstraße 24 – 26, mit einem Fußweg von 300 Meter Entfernung zur Ingenieurhochschule Dresden.<sup>304</sup>

1986 wurden die Fachrichtung Elektronische Datenverarbeitung der Ingenieurhochschule Dresden (44 Lehrende) und die Sektion Informationsverarbeitung der TU Dresden (23 Lehrende), auf Beschluss des Ministerrates der DDR an der TU Dresden als Informatikzentrum des Hochschulwesens in der DDR mit 300 Studierenden pro Jahrgang zusammengeführt (Buchwald 2008; Buchwald 2009, 1f.).<sup>305</sup> Diese an der TU Dresden konzentrierte ingenieurtechnische Ausrichtung der Informationsverarbeitung blieb strukturell unabhängig von der weiterhin existierenden mathematisch orientierten Softwareforschung der mathematischen Kybernetik und Rechentechnik an der TU Dresden.<sup>306</sup> »Es sollte

---

<sup>303</sup> Der Historiker Reiner Pommerin stellt die 3. Hochschulreform als eine umfassende Umstrukturierung dar, welche die Freiheiten der Professuren (z.B. eigenverantwortliche Einstellung von Mitarbeitern) beschnitt, mit der Absicht noch bestehende Reste »bürgerlicher« Strukturen zu beseitigen. Die Auflösung der Fachgebiete zugunsten der Einrichtung von Sektionen zielte laut Pommerin zudem darauf ab, Machtkonzentration einzelner Professoren zugunsten des Lehr-Kollektives zu beschneiden, und eine intensivere, Professuren-unabhängige Nutzung der Laboratorien, Bibliotheken und Werkstätten ermöglichen (Pommerin (Hrsg.) 2003, 1:304).

<sup>304</sup> Eine ausführliche Darstellung der Sektion Informationsverarbeitung in den Jahren 1969–1986 ist *Zur Geschichte und Lehre der Informatik an der Technischen Universität Dresden bis zur Gründung des Informatik-Zentrums 1986* (Schmidt 2008) zu entnehmen. Der Autor nimmt auch Stellung zur politischen Einflussnahme der SED auf die Arbeit der Sektion, welche hier jedoch aufgrund der Fokussierung auf das Themengebiet Datenbanken nicht ausgeführt werden – sie bezogen sich beispielsweise auf das Gebiet der Kybernetik in den späten 1950er Jahren. In den 1970ern und noch stärker den 1980ern Jahren scheint der ideologische Anteil in den Diskussionen zugunsten einer wirtschaftlichen Pragmatik gewichen zu sein, deren Steuerung jedoch fest in der Hand der SED lag.

<sup>305</sup> Geplant worden war das Zentrum in einer *Konzeption zum Aufbau des Informatik-Zentrums des Hochschulwesens an der TU Dresden*, 1986 [BArch, DR 3-1495, 2. Schicht] (zitiert nach Barkleit 2003, 269).

<sup>306</sup> Die Abteilung zur Kybernetik und Rechentechnik geht zurück auf das 1956 unter Leitung von Prof. Nikolaus Joachim Lehmann eröffnete *Institut für Maschinelle Rechentechnik* (IMR) in der Abteilung Angewandte Mathematik der Technischen Hochschule Dresden (Lehmann 1996, 128–130, 147f.; vgl. auch Sonnemann (Hrsg.) 1978, 338).

Leitungs- und Koordinierungsfunktion innerhalb des einheitlichen Bildungsprozesses in Informatik für das gesamte Hochschulwesen der DDR sowie die betriebliche Weiterbildung haben« (Buchwald 2009, 3). Damit wurde die Ausbildungsorientierung der IHD mit der Forschungs- und Ausbildungsaufgabe der TU Dresden zusammengeführt.

Die zahlreichen Umstrukturierungen deuten den Entstehungsprozess des akademischen Wissenschaftsfeld der Elektronischen Datenverarbeitung bzw. Informatik in der DDR an. Sie erfolgten ab Ende der 1960er Jahre, angestoßen durch die Bildungs- und Wirtschaftsinitiativen, welche mit dem Ziel die wirtschaftliche Situation der DDR zu verbessern, von der SED gesteuert wurden. Dieser Top-Down Logik der akademischen Umstrukturierung standen die Eigeninitiative der Lehrenden und Forschenden, das Bedürfnis nach Statuserhalt vorhandener Professuren und das Interesse an akademischer Grundlagenforschung gegenüber, beziehungsweise ergänzten sich diese auch und korrespondierten miteinander.

### 3.3.2 Forschung und Entwicklung – die Zusammenarbeit von Robotron und TU Dresden<sup>307</sup>

Daba 1600 wurde nicht allein als ein Projekt der TU Dresden entwickelt, sondern in Zusammenarbeit mit dem VEB Kombinat Robotron, welches die Praxisumsetzung begleiten sollte. Eingebettet war die Kooperation zwischen Robotron und der TU Dresden im größeren Kontext wirtschaftlicher Reformen, die ab 1962 als *Neues ökonomisches System der Planung und Leitung der Volkswirtschaft der DDR* durch das ZK der SED angestoßen wurden.<sup>308</sup> Ab 1968 sah das überarbeitete Reformkonzept *Ökonomisches System des Sozialismus*, welches Momente stärkerer Eigeninitiative (z.B. Eigenverantwortlichkeit von Betrieben bezüglich ihrer Wirtschaftlichkeit) enthielt, die langfristige Förderung und Planung volkswirtschaftlich bedeutsamer Bereiche, wie Elektrotechnik, Elektronik, Maschinenbau und weiterer vor. Ein Grund für die Reformen war der Entschluss, dass für den DDR-Sozialismus die Wissenschaft als unmittelbare Produktivkraft einen zentralen Platz einzunehmen habe. Dies zog eine Neuausrichtung von Forschung und Bildung nach sich, welche in das 1965 beschlossene *Gesetz über das einheitliche Bildungssystem* (Der Vorsitzende des Staatsrates der DDR 1965) flossen und eine stärkere Ausrichtung der akademischen Ausbildung auf industrielle Erfordernisse forcierte.

Gemeinsame Forschung und Überführung in Produkte, in der DDR als Auftragsforschung bezeichnet, galt neben Betriebspraktika der Studierenden als ein wichtiges Steuerungsmittel, um die industriennahe Forschung zu bestärken (Pommerin (Hrsg.) 2003, 1:293ff.). Die Auftragsforschung wurde nun zwischen den Sektionen und Betrieben direkt ausgehandelt, wobei sich die SED eine Steuerungsfunktion vorbehielt: »Die Profilierung der [Auftrags-]Forschung nimmt in der TU bezeichnenderweise jetzt die Kreisleitung der SED [der TU Dresden] vor; sie achtet darauf, daß sich diese nach der gesellschaftlichen Aufgabenstellung richtet« (ebd. 311).

Wie kam das Forschungsprojekt über relationale Datenbanken letztlich konkret zu Stande? Schließlich gab es eine Reihe von Zweifeln, ob ein solches Vorhaben angesichts der begrenzten Mittel realistisch sein würde. Die Zusammenarbeit zwischen dem VEB Robotron und verschiedenen Sektionen der TU

---

<sup>307</sup> Die Rahmenbedingungen der Forschungsk Kooperation aus Sicht der Beteiligten wurden bereits detailliert in dem Text *Technologietransfer am Beispiel von DABA 1600* (Demuth/Wiggert 2008, 283) dargestellt.

<sup>308</sup> Abele ordnet in *Modernisierung der Industriegesellschaft – Hochschulpolitik in der DDR* die Restrukturierungen des Hochschulwesens in die an der Industrieproduktion orientierte Wirtschaftspolitik der SED ein (Abele 2003, 173–175).

Dresden legten die Beteiligten in einer institutionellen Vereinbarung, einer sogenannten Komplexvereinbarung vom 6. Dezember 1977 fest.<sup>309</sup> Diese verzeichnete als Ziele, den »wissenschaftlich-technischen Fortschritt zu beschleunigen, planmäßig neueste, dem wissenschaftlich-technischen Höchststand entsprechende Ergebnisse« zu erreichen und »ihre schnelle Überführung sowie Anwendung in der Produktion zu sichern«, sowie »eine hochqualifizierte und praxiswirksame Ausbildung und Weiterbildung der Kader« zu gewährleisten (Sektion Informationsverarbeitung 1977, Fol 2). Die konkreten Themen der Zusammenarbeit fixierte man in einer Forschungskonzeption, welche die Jahre 1981–1985 umfasste und die zwischen verschiedenen Ebenen der Universität unter Leitung des Direktorat Forschung mit Robotron diskutiert wurden.<sup>310</sup>

Vorgeschlagen wurden in der Vorbereitung 1979 seitens der Sektion 08 die Themen Programmiermittel für SKR (Prof. Loeper), Produktionssteuerung (Prof. Stahn), Strukturierte Programmierung und Daten- und Programmbanken (Prof. Schubert). Konkret schlug Schubert vor, sich im Zuge dieses Punktes mit einer »Datenmanipulationssprache auf COBOL-Basis für Kleinrechner«, einem »Relationalen Datenbanksystem für ESER«, »Technologie der Programmbanksysteme und experimentelle[n] Entwicklung einer Programmbank am Beispiel der mathematischen Statistik« auseinander zu setzen (Bode 1979, 1–2; vgl. Liebscher 1978). Schuberts Vorschlägen gingen offensichtlich interne Diskussionen der *Zeitweiligen Arbeitsgruppe zu Datenbankforschung* beim Programmrat an der Akademie der Wissenschaften der DDR voraus, von denen jedoch keine Archivalien überliefert sind. Möglicherweise diskutierte Schubert dort mit Kollegen den Focus auf die relationale Datenlogik. Außerdem erfolgte eine Auseinandersetzung mit dem Thema *Untersuchungen der Prinzipien der Organisation von Informationsbasen* in der Mehrseitigen Regierungskommission Rechentechnik (MRK), welche für das ESER und SRK-Programm zuständig war (Sektion Informationsverarbeitung 1978, Fol 1).

Innerhalb des ZFT Robotron diskutierte man anfangs die Idee, ein relationales Datenbanksystem für Kleinrechner zu erarbeiten kontrovers: »Eine Datenbank auf einem Kleinrechner betreiben zu wollen, wie er z.B. damals mit den Anlagen der Familie Robotron K 1600 zur Verfügung stand – Ende der siebziger Jahre schätzte man das zunächst als ein durchaus ungewöhnliches Vorhaben ein. Zumindest zweifelten die Entwickler des für die Großrechner (R 300 bzw. die nachfolgenden ESER-Rechner) ausgelegten Datenbankbetriebssystems DBS/R im damaligen Zentrum für Forschung und Technik (ZFT) des Kombinat Robotron den Erfolg eines solchen Vorhabens stark an« (Demuth/Wiggert 2008, 280).

Trotzdem kam es zur Zusammenarbeit, da man für die K 1600 Kleinrechner Software benötigte. Zur Vereinbarung zwischen ZFT Robotron und TU Dresden gehörte nicht nur allein, die Forschung auf diesem Gebiet zu betreiben, sondern auch deren Realisierung und Überführung in Softwareanwendungen. Im Zuge der ersten Stufe der Auftragsforschung erarbeitete man ein Pflichtenheft, welches die konkreten Aufgaben formulierte. Man einigte sich darauf, dass Robotron die untere Schicht des Datenbankbetriebssystems bereitstellen würde und die TU als Aufsatz den Compiler und die

---

<sup>309</sup> Als gesetzliche Grundlage wurde die *Verordnung über die Leitung, Planung und Finanzierung der Forschung an der Akademie der Wissenschaften und an den Hoch- und Fachschulen* vom 23.8.1972 genannt (Sektion Informationsverarbeitung 1977, Fol 3). Komplexverträge schloss die TU Dresden mit zahlreichen Industriebetrieben der DDR, aber auch mit der Stadt Dresden ab.

<sup>310</sup> Eingebettet war dies in den 1980er Jahren in eine ganze Reihe von Beschlüssen zur Intensivierung der Beforschung von Mikroelektronik, die der Physiker und Historiker Gerhard Barkleit aufführt: *Beschluss des Politbüro des ZK der SED über Aufgaben der Universitäten und Hochschulen in der entwickelten sozialistischen Gesellschaft* vom 18.3.1980, *Festlegungsprotokoll der Beratung des Ministeriums für Wissenschaft und Technik mit dem Ministerium für Hoch- und Fachschulwesen zu den Aufgaben für den Staatsplan Wissenschaft und Technik 1986, sowie 1986–1990* vom 23.8.1985 (zitiert nach Barkleit 2003, 260–264).

SQL-Optimierungskomponente. Finanzielle Leistungen im Zuge des Vertrags erfolgten nicht, jedoch konnten Rechenzeit im Robotron-Rechenzentrum Dresden für die TU Dresden und die Bibliotheken und Informationsdienste gegenseitig genutzt werden.

Laut einem TU-internen Bericht von 1981 plante man die Grundlagenforschung in den Stufen G1 und G2 ab Juni 1981. Unter dem Titel *Beiträge zum Entwurf und Implementierungskonzeption von Datenbankbetriebssysteme mit Mehrebenenarchitektur* sollte die Forschungsstufe BES (2), d.h. den Weltstand mitbestimmend, erreicht werden. Die Ergebnisse der Grundlagenforschung übergab man dem ZFT Robotron in Form von Zwischenergebnissen.<sup>311</sup>

Die Anwendungsforschung in den Stufen A1 bis A4 legte man vom September 1979 bis März 1983 unter dem Titel *Implementierung eines relationalen Datenbankbetriebssystems mit Datenbanksprache SE-QUEL für K 1630* auf der Forschungsstufe ENT fest.<sup>312</sup> Die Ergebnisse sollten dem internationalen Stand entsprechen. Das Forschungskollektiv strebte ein »Spitzenergebnis« an und begründete dies mit der »Senkung des Programmieraufwandes«, der »Komplettierung der Rechner des Typs K1630« sowie der »Sicherung des Vorlaufs für effektiven Export« (Sektion 08 1982, Fol 1). Der Bericht erklärt auch, warum überhaupt ein Bedarf an einem eigenen relationalen Produkt gesehen wurde: Wollte man ins westliche Ausland exportieren, musste dies mit lizenzrechtlich unbedenklicher Softwareausstattung erfolgen.

Neben vertraglichen und planerischen Vorgaben erweist sich der rege Austausch zwischen beiden institutionellen Akteuren als eine weitere Voraussetzung für das Zustandekommen von DABA 1600. So arbeiteten viele Informatikprofessoren zuvor oder parallel bei Robotron. Die Firma rekrutierte hochqualifizierte Absolventen der TU als Arbeitskräfte. Dies ist anhand von DABA 1600 exemplarisch nachzuvollziehen: Prof. Schubert beispielsweise war seit 1969 Hauptabteilungsleiter im Forschungszentrum des VEB Robotron, ab 1970 Honorarprofessor an der TU und ab 1972 ordentlicher Professor an der TU Dresden (Petschel (Hrsg.) 2003).

Nach Ende der Forschungsarbeiten und des Studiums an der TU wechselten einige Projektteilnehmer zum DABA 1600-Team bei Robotron und gingen mit der Erfahrung dieses ersten relationalen DBMS um 1987 in das neue Schwerpunktprojekt INTERBAS über. Die Anforderungsveränderung von DABA 1600 zu INTERBAS erfolgte nicht zufällig, denn bereits 1981 vermerkte Prof. Schubert im hochschulinternen Bericht *Planentwurf Forschung 1981–85. Profilierung auf Schwerpunktaufgaben und Leistungserhöhungen in Auswertung des X. Parteitages der SED*: »Die Plandiskussion in Verbindung mit den Ergebnissen der E3-Verteidigung führte zu einer wesentlichen qualitativen Erweiterung der Aufgabenstellung (Unterstützung von Netzwerk-Datenstrukturen neben den ursprünglich vorgesehenen ausschließlich relationalen Strukturen durch die zu entwickelnde und implementierende

---

<sup>311</sup> Die Grundlagenforschung in den Stufen G1 bis G4 war auf maximal vier Jahre angelegt, Staatsplanthemen in den Stufen Auftragsforschung A1 bis A4 für 3 Jahre und Entwicklung in den Stufen E1 bis E5 für 2 Jahre (Gräßler 1972, 28), festgelegt in *Anordnung über die Durchführung von Verteidigungen wissenschaftlich-technischer Aufgaben und Ergebnisse* vom 23.5.1973) und *Anordnung über die Nomenklatur der Arbeitsstufen und Leistungen von Aufgaben des Planes Wissenschaft und Technik* vom 28.5.1975.

<sup>312</sup> Ihre Einschätzung als ENT und BES(2) basierte auf einer offiziellen Nomenklatur für Forschungs- und Entwicklungsarbeiten, die für universitäre und betriebliche Forschungsinstitutionen galt. Diese Stufen fragte die Staatlichen Zentralverwaltung für Statistik in jährlichen Formularen ab: »BES (1) – der fortgeschrittene intern. Stand wird bestimmt, BES (2) – der fortgeschrittene intern. Stand wird mitbestimmt, ENT – die Ergebnisse entsprechen dem internationalen Stand, NER – der internationale Stand wurde nicht erreicht, SPEZ – spezielles Problem, intern. nicht vergleichbar« (Direktor für Forschung 1981, Fol 1). Aus solchen Einschätzungen erstellte die Staatlichen Zentralverwaltung für Statistik Berichte an die Ministerien und die SED-Leitungsebenen über die Erfüllung, Übererfüllung oder Nichterfüllung der geplanten Forschungsleistungen.

Datenbanksprache)« (Schubert 1981, Fol 3). Genau diese Anforderung entsprach dem späteren Projekt INTERBAS, und es ist davon auszugehen, dass sie in der Diskussion zwischen dem ZFT Dresden und der TU Dresden entstanden.

### 3.3.3 Ein Problemorientiertes Seminar als Knotenpunkt des Wissenstransfers

Die vorliegende Arbeit untersucht im Unterschied zum Konferenzbeitrag der Autoren Demuth/Wiggert *Technologietransfer am Beispiel von DABA 1600* (2008) stärker die kontextuelle Einbettung in die verschiedenen Ordnungen des Wissens. Dazu wurde intensiv die heute verfügbare Literatur, insbesondere Dissertationen, Zeitungsveröffentlichungen und Softwaredokumentationen gesichtet, sowie Interviews mit den beteiligten Akteuren geführt. Wie und unter welchen Bedingungen fand der Wissensaustausch statt?

Im Jahr 1973 traf sich auf Einladung von Prof. Karl-Heinz Müller (Datenbanken), der neben Prof. Dietrich Schubert (Systemmodellierung) die zweite Professur im Fachgebiet Datenbanken als Teil des Wissenschaftsbereiches Automatisierte Informationssysteme an der TU Dresden inne hatte, erstmals ein *Problemorientiertes Seminar Datenbanken*.<sup>313</sup> Das Seminar war nicht allein den Studierenden und Lehrenden zugänglich, sondern auch den »Mitarbeitern der Entwicklungskollektive des VEB Robotron ZFT, Projektanten und Anwendern« (Müller 1975, 44). Ein ähnlich aus Forschern, Herstellern und Anwender zusammengesetztes Seminar diskutierte im Folgenden Jahr unter anderem das dateorientierte Datenbanksystem BASTEI, das am Netzwerkmodell orientierte Robotron DBS/R und das Dokumentenrecherchesystem Robotron AIDOS.

Karl-Heinz Müller stellte einen »formalen Beschreibungsapparat« basierend auf der »Mengenlehre« vor – in der DDR die möglicherweise erste belegbare Bezugnahme auf das relationale Kalkül von Edgar F. Codd (ebd.). Dies bedeutet nicht notwendigerweise die Kenntnis der Originaltexte von Codd. Aber es war das gleiche Jahr, in dem im Zuge der National Computer Conference in Anaheim/USA zum ersten Mal ausführlicher, d.h. in mehreren Sessions die relationale Algebra diskutiert und publiziert wurde (Codd/Edgar F. 1975). Ebenfalls 1975 erfolgte die Publikation einer hitzigen Diskussion zwischen Befürwortern der relationalen Algebra und Protagonisten des netzwerk-orientierten Modells, aus dem Jahr zuvor (Codd/Edgar F./Bachman 1975). Auch wenn es heute im Einzelnen nicht mehr belegt werden kann, wäre es verwunderlich, wenn die Fachleute in der DDR diese Diskussion überlesen hätten.

Für das Problemseminar ein Jahr später, 1976, ist die Diskussion der Codd'schen Thesen direkt belegt, die gemeinsam und vergleichend mit dem CODASYL-Netzwerkmodell erörtert wurden (Müller 1977). Im gleichen Jahr war Müllers Kollege, Prof. Dietrich Schubert, zu Gast in Moskau und sprach dort zu einer Tagung des Rates der Chefkonstruktoren des ESER mit dem Schwerpunkt Datenverwaltung über *das Relationenkonzept als theoretische Grundlage für den Entwurf von Datenstrukturen und Datenbanksprachen*.

Vermuten lässt sich, dass Karl-Heinz Müller die relationale Algebra nicht stärker weiterverfolgte, da es Mitte der 1970ern für eine Anwendungspraxis noch nicht ausgefeilt genug war. Dies korrespondiert mit dem Umstand, dass er neben seiner Professur, seit 1969, auch als Leiter des VEB Robotron

---

<sup>313</sup> Problemorientierte Seminare fanden mit verschiedenen Themenschwerpunkten in der gesamten Sektion Informationsverarbeitung als Teil der Weiterbildungsarbeit statt (Schmidt 2008, 151).

Rechenzentrums Dresden tätig war (Petschel (Hrsg.) 2003, 3:660) und damit unmittelbar in Anwendungsfragen involviert. Sein Kollege Prof. Dietrich Schubert hingegen griff die relationale Algebra verstärkt auf, wie sich aus einer Reihe von Forschungsarbeiten seiner Studierenden und eigenen Vorträgen ablesen lässt. Insofern ist hier eine Differenzierung der beiden Professuren zu erkennen.

1979 und 1982 fanden die Problemseminare durch beide Professuren gemeinsam statt, wobei bereits 1979 eine Dissertation zum relationalen Prototypen REDAMS thematisiert wurde, angebunden an die Professur Schubert. 1982 arbeiteten die Studierenden und Mitarbeiter, betreut durch Schubert, zur relationalen Datenbanksprache SEQUEL für das Projekt DABA 1600 und diskutierten diese im Seminar. Zugegen waren auch Vertreter des ZFT Robotron, die ihren Arbeitsstand zum Betriebssystem von DABA 1600 darstellten. Rolf Heinemann, Abteilungsleiter der Gruppe »Entwicklung von Software für Speicherung und Wiederauffindung formatierter Dateien«, stellte für das ZFT Robotron die weitere Strategie des netzwerk-orientierten DBMS DBS/R und des Dokumentenrecherchesystems AIDOS vor. Für dieses Seminar ist eine Reihe von Anwendungsberichten verzeichnet, zum Beispiel einer Fertigungsdatenbank bei Carl Zeiss Jena, einer Rechnungs- und Statistikdatenbank und einer weiteren Datenbank, welche an ökonomische Planungssysteme angebunden werden sollte (Müller 1982).

Für das Folgejahr, 1983, konstatierte Müller in der Zeitschrift *rechentechnik/datenverarbeitung*:

»Besonders zeigte sich, dass sich bereits eine breite Anzahl von Datenbanksystemen in der Anwendung befinden und das einheitliche, universelle DB-System nicht existiert. Hauptursache dürfte sein, daß die Anwendungsbedürfnisse doch stark differieren [...]« (Müller 1984). Deutlich wird hier eine gewisse Enttäuschung, hatte man doch weltweit gehofft, mit elektronischen Datenbanken zentralisierte und einheitliche Speicherlösungen zu schaffen, welche die Datenhaltungsprobleme ein für alle Mal erledigen. Es vermittelt sich jedoch auch das Bild einer diversifizierten Softwarelandschaft der Datenbankmanagementsysteme in der DDR.

Diese Verdichtung zeigt auch, welche Bedeutung das *Problemorientierte Seminar Datenbanken* im institutionellen Netzwerk für elektronische Datenbanken in der DDR und speziell des relationalen Ansatzes einnahm. Es vereinte Akteure aus Anwendung, Entwicklung und Forschung als Plattform interinstitutionellen und interdisziplinären Austausches. Die relationale Algebra war nur eines von mehreren Forschungsthemen und der Erfolg zu jenem Zeitpunkt keineswegs absehbar. In der Anwendungspraxis spielte die relationale Algebra noch eine unwesentliche Rolle, und der Schwerpunkt in den Jahren 1975–1985 lag auf der grundlegenden Erforschung. Damit unterscheidet sich der Stand der Produktivsysteme in der DDR nur unwesentlich vom Westen. IBM führte mit dem Prototypen System R (1974–1978) den Forschungsstand international unangefochten an. Es gelang Ingres, Oracle und auch IBM, ab Anfang der 1980er, relationale Produktivsysteme zu etablieren.

Dafür gibt es Gründe: Ebenso wie die zugrundeliegende Rechentechnik handelte es sich bei den Datenbanksystemen um Infrastruktur. Diese, einmal installiert, war zeitlich äußerst persistent. Somit löste nicht eine Logik der Datenorganisation die nächste ab, sondern sie koexistierten über Jahrzehnte, je nachdem, zu welchem Zeitpunkt die Arbeiten an einer Anwendung begannen.

### 3.3.4 Forschungsergebnisse aus der Sektion Informationsverarbeitung

Nachdem der allgemeine Aufbau der Forschung dargestellt wurde, soll nun deren konkreter Ablauf fokussiert werden. Dies bietet nicht nur eine historische Perspektive, sondern auch eine konzeptionelle Neubetrachtung des relationalen Konzepts aus Perspektive seiner ostdeutschen Adaption. Der folgenden Abschnitt untersucht heute zugängliche wissenschaftliche Veröffentlichungen, verbunden mit zwei

Hauptfragestellungen: 1.) Welche relevanten Quellen werden angegeben, welche Wege nahm der Wissenstransfer im konkreten Fall? 2.) Welche Begründungen werden angegeben für die Erforschung der relationalen Algebra und führte diese zu einer spezifischen Entwicklung ›sozialistischer‹ Technologie? Um diese Fragen zu beantworten, ist es notwendig einen detaillierten Blick in die Entwicklungen zu werfen.

#### 3.3.4.1 Vorarbeiten

Ende 1977 publizierte Prof. Dietrich Schubert von der TU Dresden in *edv-aspekte*, einer regelmäßigen Beilage zur Zeitschrift *rechentechnik/datenverarbeitung*, einen Überblicksartikel zu *Stand und Perspektiven der Entwicklung von Datenbankbetriebssystemen* (Abb. 69). Schubert setzte sich mit der Rolle der Datenbanksysteme in der Informationsverarbeitung auseinander. Er klassifizierte und bewertete Datenbanksysteme, indem er deren Aufgabenstellung und Merkmale diskutierte und dann die im Rahmen von ESER verfügbaren Figurationen einschätzte. Zu den grundlegenden Merkmalen von Datenbanksystemen zählte er unter anderem:

- »Variabilität der grundlegenden Informationseinheit«, die damals noch neue Möglichkeit der Verarbeitung verschiedener Datenformate, z.B. Bilder, Text, unstrukturierte Zeichenketten in einer gemeinsamen Datenstruktur, der Datenbank,
- »Variabilität der Datendarstellung«, die Unabhängigkeit der Datendarstellung von den maschinellen Gegebenheiten der Hardware,
- »Adaptierbarkeit«, das Potenzial, vorhandene Softwarefigurationen für später erkannte Bedürfnisse anzupassen und zu erweitern,
- »Datenunabhängigkeit«, vor allem die Unabhängigkeit der Zugriffspfade zu den Speicherorten der Daten betreffend,
- »Datenschutz«, welcher in Abweichung von der heutigen Bedeutung die Sicherung der Zugriffsrechte bedeutete – das Recht zu lesen und schreiben,
- Fragen der »Datensicherheit« – die Möglichkeit der Protokollierung, von Backups und Wiederherstellung nach Abstürzen.

Das Besondere an dieser Klassifizierung ist, dass Schubert damit den Lesern der Zeitschrift – größtenteils DDR-Fachleute und Spezialisten, die an Großrechnern arbeiteten – ein akademisch fundiertes Handwerkszeug zur Verfügung stellte, um gegebenenfalls einzusetzende Datenbanksysteme abschätzen zu können. Im zweiten Teil des Artikels diskutierte er zukünftige Entwicklungen und stellt grundlegend die relationale Algebra vor, um dann über die Drei-Ebenen-Architektur des ANSI-SPARC Modell zu schreiben.

Schuberts Text stellt in weiten Teilen die Reaktion auf eine ›westliche‹ Quelle dar, denn er bezog sich stark auf Hartmut Wedekinds in Mannheim/Wien/Zürich herausgegebenes Kompendium *Datenbanksysteme*, Band 1, von 1974.<sup>314</sup> Der erste Teil seines Textes gibt Wedekind in anderer Reihenfolge und besser strukturiert als das Original wieder. In seinem zweiten Teil verkündet Schubert die frohe Kunde von der Ankunft der relationalen Algebra in der DDR-Fachgemeinde. Wedekinds Buch muss eine zentrale Rolle in der Wissensaneignung in Dresden gespielt haben, denn es wird auch von Birgit Demuth,

---

<sup>314</sup> Bd. 2 gab Wedekind gemeinsam mit Theo Härder heraus, der in Folge seiner Forschungsarbeiten an System R in Kalifornien, seine Dissertation *Das Zugriffszeitverhalten von relationalen Datenbanksystemen* 1975 an der TU Darmstadt im Fachbereich Informatik verteidigte. Dieser zweite Band wurde in der DDR ebenfalls intensiv rezipiert.



die ebenfalls bei Prof. Schubert studierte, erwähnt: »Das Buch wonach ich das eigentlich gelernt habe, war von Professor Wedekind aus Nürnberg« (Interview Demuth 2018, Fol 2) und ist auch bei allen weiteren Dissertationen als Quelle vermerkt.<sup>315</sup>

Kurzbezeichnung Betriebssystem	Endnutzerklassen	Endnutzersichten	Datenunabhängigkeit	Art der Bedienungsfunktion	Charakteristik der DMS
SAWI DOS/ES	Anw. progr., Fachspr.- Nutzer	Menge von Sätzen (Werte-n-Tupeln)	Datentyp Satzstruktur	Recherchesystem	eigenständig
BASTEI DOS/ES	Anw. progr., param. Nutzer	Netzwerkstruktur	Datentyp	Operationelles System	Trägersprache (Makros)
AIDOS DOS/ES und OS/ES	Fachspr.- Nutzer param. Nutzer	inverse Struktur <sup>*)</sup>	Datentyp Satzstruktur Zugriffspfad <sup>*)</sup>	Recherchesystem	eigenständig
DASY DOS/ES	Anw. progr.	Matrixstruktur	Datentyp	Operationelles System	eigenständig, Trägersprache (Unterprogr.)
DBS/R OS/ES	Anw. progr., Fachspr.- Nutzer	Netzwerkstruktur, inverse Struktur <sup>*)</sup>	Datentyp Satzstruktur	Operationelles System Recherchesystem <sup>*)</sup>	eigenständig (Einfügung von Moduln); Trägersprache (Makros)
EDO OS/ES	Anw. progr.	Matrixstruktur	Datentyp	Operationelles System	Trägersprache (Unterprogr.)

<sup>\*)</sup> mit Einschränkungen

Abb. 69: Datenbanksysteme des ESER 1977 laut Schubert (Schubert 1977, 6).

Die starke Bezugnahme auf Datenbanksysteme von Wedekind 1974, zeigt die Notwendigkeit an, dieses als Quelle näher in Augenschein zu nehmen. Die Monographie entstand aus einer Reihe von Vorlesungen, welche er im Sommer 1974 an der TU Darmstadt zum Schwerpunkt Datenbanken hielt und welche als Teil I eines zweibändigen Kompendiums erschienen. Mangels geeigneter Lehrmaterialien fasste Wedekind diese zu einem Lehrbuch zusammen. »Dem Wedekind« ist in den Dissertationen der TU Dresden Datenbank-Forschungsgruppe zahlreich zu begegnen. Die Verwendung war klar und wissenschaftlich redlich gekennzeichnet und das unterschied die Forschung maßgeblich von den Adaptionen von Software in der DDR, welche die Herkünfte zu verschleiern suchten.

In seiner Einleitung setzt Wedekind den Schwerpunkt auf die relationale Algebra, da sie »sich wegen seiner Einfachheit in der Praxis gegenüber den anderen Modellen in Zukunft durchsetzen wird« (Wedekind 1974, 1:5) und nahm auch Bezug auf die Auseinandersetzungen zwischen Vertretern relationaler und hierarchischer Ansätze, die in seiner Wahrnehmung »wenn auch scherzhaft«, so doch als »Religionskrieg« geführt würden (ebd.). In diesem ersten Band widmete er sich Fragen der Architektur von Datenbanksystemen, der Modellierung, spezifisch der relationalen Datenlogik und relationalen und nicht-relationalen Datenmanipulationssprachen (Abfragesprachen). Hartmut Wedekind führte in

<sup>315</sup> Wedekinds Publikation ist in der 1974er Ausgabe in der SLUB einmal vertreten und ebenfalls in der 2. Auflage von 1981. Ein Nachweis für die Auflage von 1976 mit Hilfe von Worldcat für das Gebiet der ehemaligen DDR verzeichnet das Buch im Bibliotheksbestand der Universität Halle, der Universität Weimar, der Humboldtuniversität Berlin, der Hochschule Ilmenau, sowie der deutschen Nationalbibliothek Leipzig. Für die spätere Implementierung von DABA 1600 wurde laut Aussage von Demuth verstärkt Theo Härder's Monographie *Implementierung von Datenbanksystemen* (Härder 1978) herangezogen.

seinem Literaturverzeichnis die relevanten, meist englischsprachigen Publikationen seiner Zeit auf, mit Schwerpunkt auf IBM Research Reports und den Communications und Proceedings of the ACM.<sup>316</sup> Wedekinds Kompendium war nicht allein eine technologische Auseinandersetzung sondern diskutierte explizit strategische Fragestellungen. Herausragend ist hier ein Zitat, in dem Wedekind den 1974 aktuellen Stand einschätzte: »Hierarchische Systeme und Netzwerksysteme sind zu kompliziert und zu restriktiv. Die Systeme sind vom Methodologischen her nur schwer überschaubar. Logische Gesichtspunkte und Wirksamkeitsargumente werden permanent bewußt und unbewußt miteinander vermengt. Man wird nach Ansicht des Verfassers aber erst dann auf »einer breiten Front« zu einem Relationen-System übergehen, wenn die gewaltigen Investitionen in das System IMS II<sup>317</sup> abgeschlossen sind oder wenn sich in der CODASYL-Gruppe die Erkenntnis durchsetzt, daß es einen besseren Weg gibt« (Wedekind 1974, 1:181). Eine derart klare Einschätzung dürfte nicht ohne Resonanz in der Rezeption an der TU Dresden geblieben sein.

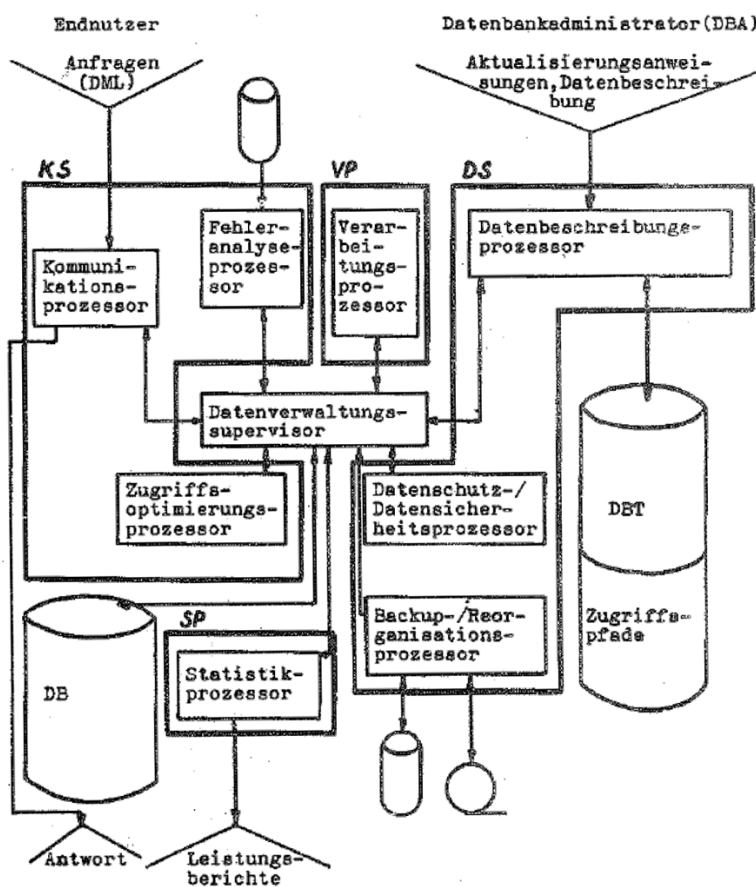


Abb. 70: »Architektur eines relationalen Datenbanksystems in topographischer Sicht«, Administratoren und Endnutzer greifen auf verschiedene topologische Aspekte des DBMS zu (Wenk 1978, 3).

Nach der breitenwirksamen Publikation Prof. Müllers in *edv-aspekte* erschien 1978 von seiner Mitarbeiterin Vera Wenk, in der Schriftenreihe der TU Dresden, eine Publikation mit dem Titel *Zur Implementierung des Relationenkonzeptes für ESER*. Sie diente Überlegungen zur Architektur eines

<sup>316</sup> In der DDR hat es zwar auch Zugriff auf die Communications of the ACM gegeben, IBM Research Reports scheinen als Primärliteratur jedoch nicht zur Verfügung gestanden zu haben.

<sup>317</sup> Das IBM Information Management System II verwendete eine hierarchische Datenbanklogik.

relationalen Datenbanksystems für Großrechner: »Relationale DBS [Datenbanksysteme – F.H.] sind in den letzten Jahren Gegenstand heftiger Diskussionen geworden, haben in Forschung und Entwicklung starke Beachtung gefunden und es wurden bisher ca. 30 Implementierungen vorwiegend experimenteller Art realisiert«, rahmte Wenk ihre Untersuchung, die neben den öffentlichen Vorträgen Schuberts den Beginn der Forschungen markiert (Wenk 1978, 2). Recht unvermittelt, vermutlich der Beschränkung auf 16 Seiten geschuldet, erörterte Wenk dann die grundsätzliche Konzeption für eine Systemarchitektur.

Ihr Entwurf bestand aus einer Reihe von Modulen (Abb. 70). Ein ›Kommunikationsprozessor‹ optimierte die Nutzer-Anfragen und übersetzte in eine maschinennahe Sprache. Der ›Datenbeschreibungsprozessor‹ diente der Strukturierung der Datenbank, also beispielsweise dem Hinzufügen von Spalten oder Tabellen (Relationen). Ein ›Statistikprozessor‹ sollte der Optimierung der Zugriffspfade auf dem Speichermedium dienen und Indexe und Verteilung der Daten auf der Festplatte bei Bedarf neu organisieren. Ein zentraler ›Datenverwaltungssupervisor‹ war die automatisierte Schnittstelle, welche die Arbeit der einzelnen Module untereinander koordinierte (Wenk 1978, 3). Darauf aufbauend erörterte sie mit Hilfe der Graphentheorie die optimale Vermittlung zwischen Nutzeranfragen, den daraus resultierenden Zugriffspfaden und deren Übersetzung in Assembler/Maschinensprache.

In der Literatur griff sie auf Vorträge Prof. Dietrich Schuberts zurück, die er auf internationalen Fachkonferenzen gehalten hatte, auf einschlägige Artikel aus den *Communications of the ACM* und auf Hartmut Wedekinds Kompendium *Datenbanksysteme*, Bd. 1, von 1974. Retrospektiv ist an dem Literaturverzeichnis Wenks auffallend, dass es in Bezug auf ›westliche‹ Quellen keinerlei Berührungspunkte gab. Gleiches lässt sich für die anderen hier untersuchten Literaturverzeichnisse konstatieren. Bundesrepublikanische und US-Literatur waren einträchtig neben DDR und Literaturangaben aus dem RGW aufgeführt.

Kurz nach Vera Wenks erster Veröffentlichung, im April 1978, erschien ebenfalls in der Schriftenreihe der TU Dresden vom Dozenten Wolfgang Belke und den Doktoranden Bernd Keller und Michael Staruß das 16-seitige *Konzept einer COBOL-orientierten Datenmanipulationssprache REDAMS für Kleinrechner*. Im Unterschied zu Wenk erkundete das Trio die Möglichkeiten nicht für Großrechner, sondern für Kleinrechner: »Mit der Entwicklung und Produktion von Kleinrechnern wurde und wird [...] die Möglichkeit geschaffen, ein breites Spektrum von Betrieben und Institutionen mit eigener Rechenkapazität zu versehen« (Keller/Starruß/Belke 1978, 2). In der Einleitung wurde unmittelbar auf Edgar F. Codd verwiesen, der jedoch in den Literaturangaben nicht verzeichnet ist und zu diesem Zeitpunkt vermutlich allein aus Sekundärquellen bekannt war. Es findet sich hingegen Chris Date, ein enger Kollege Edgar F. Codd, mit einem Text in der populären Computerzeitschrift *Datamation*, welcher wiederum auf Material aus seinem 1975 erschienenen Buch *An Introduction to Database Systems* aufbaute (Abb. 71). Neben einer allgemeinen Einführung in die relationale Algebra, demonstrierte Date drei relationale Operatoren: Select, Project und Join. Date argumentierte zusammenfassend, dass es im Computing einen klaren Trend zu einer besseren Usability gäbe, und dass die relationale Algebra im Zuge dessen eine zunehmende Rolle spielen werde (Date 1976, 52), eine Argumentation, der sich Keller/Starruß/Belke anschlossen.<sup>318</sup>

---

<sup>318</sup> *Datamation* war kein akademisches Magazin, es diente zur allgemeinen wirtschaftsorientierten Information zur elektronische Datenverarbeitung, ähnlich der in Deutschland bekannten Zeitschrift *c't* (Magazin für Computertechnik). Die englischsprachige, monatliche Zeitschrift umfasst in den 1970er Jahren zwischen 100 und 200 Seiten von Übersichtsartikeln, mit zahlreicher Werbung für neueste Produkte und wurde durch die Universitätsbibliothek als Printausgabe im Abonnement seit 1969 bezogen.

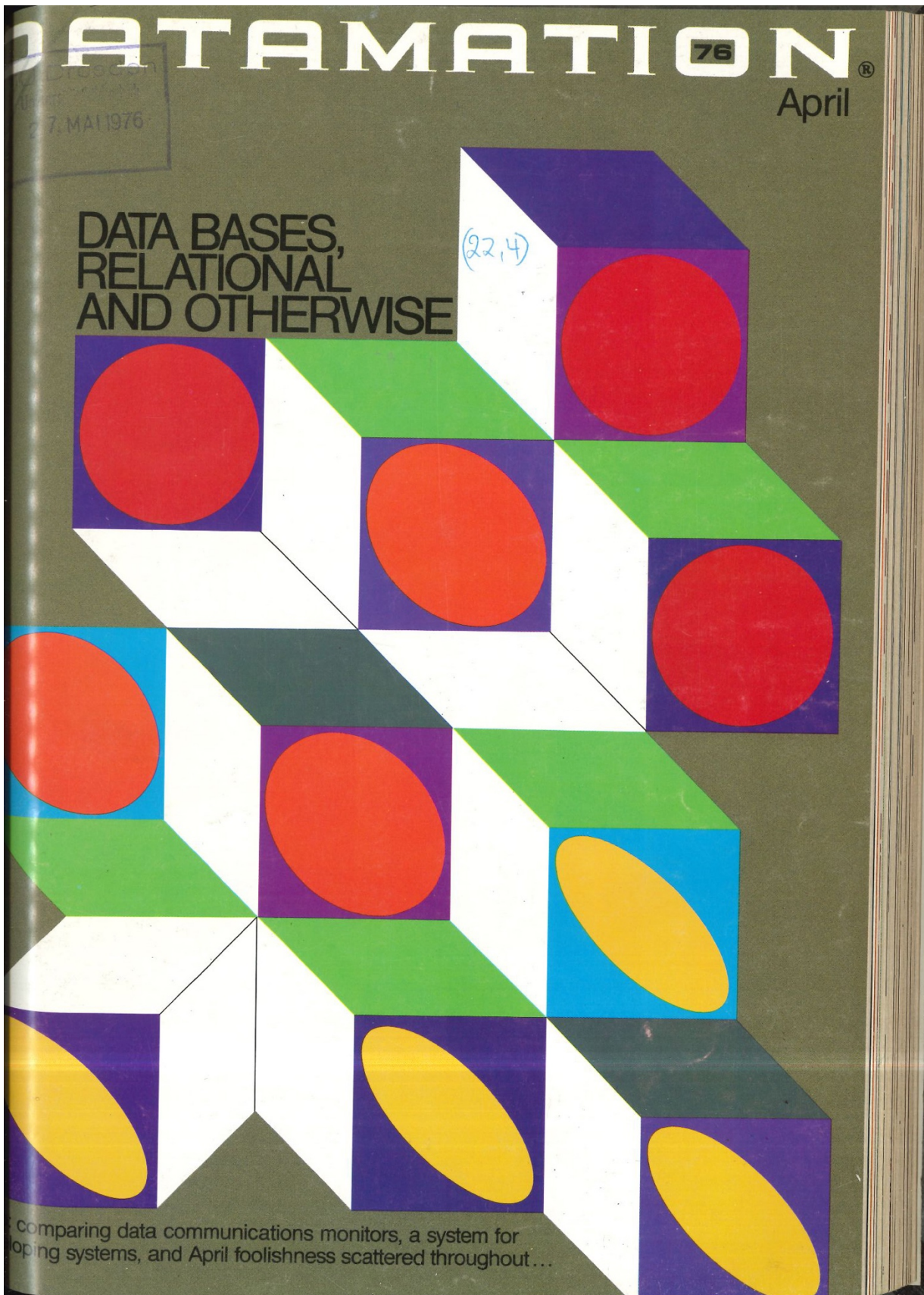


Abb. 71: Cover der Datamation Ausgabe mit dem Schwerpunkt *Databases, Relational and Otherwise* (Datamation 1976).

Die von Keller/Starruß/Belke vorgeschlagene Abfragesprache REDAMS (RELationale DATen Manipulations Sprache) sollte auf die verbreitete Sprache COBOL aufsetzen. Suchanfragen würden durch einen Pre-Compiler in ein ablauffähiges COBOL-Programm übersetzt werden, welches wiederum durch den COBOL-Compiler in ausführbaren Assembler/Maschinencode übersetzt würde (Abb. 72). Darin ähnelte der Ansatz dem Vorschlag von Vera Wenk. Ziel dieser Architektur war es, die angestrebte Maschinenunabhängigkeit zu realisieren, da COBOL für die meisten der vorhandenen Betriebssysteme zur Verfügung stand und REDAMS auf COBOL aufbaute.

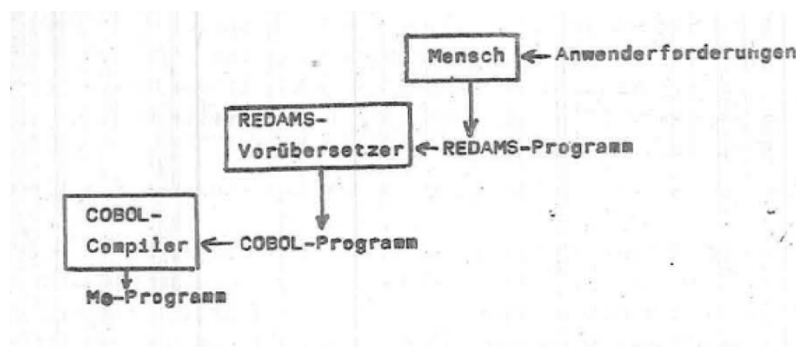


Abb.1: Grobblauf der Arbeit mit REDAMS

Abb. 72: REDAMS: Der Vorübersetzer (engl: Pre-Compiler) überträgt die Anfrage in die Hochsprache Cobol. Der Cobol-Compiler wiederum übersetzt die nunmehr in Cobol vorliegende Anfrage in ein maschinen-orientiertes Programm, in Assembler oder Maschinencode (Keller/Starruß/Belke 1978, 3).

Außerdem ermöglichte es dieser Ansatz, REDAMS auf einem der Universität im Robotron-Rechenzentrum zur Verfügung stehenden Großrechner zu testen und debuggen, noch bevor überhaupt einer der Kleinrechner K 1600 zur Verfügung stand. In Anpassung an die projektierten Möglichkeiten und Beschränkungen des K 1600 reduzierten sie gegenüber Codd's ursprünglich vorgeschlagenen vier Operationen, Projektion, Verbund, Restriktion/Selektion, den operationalen Umfang auf ein für die gegebenen Ressourcen angemessenes Maß.

Im gleichen Zeitraum, 1978, absolvierte die Studentin Birgit Demuth<sup>319</sup> ihr obligatorisches, einsemestriges Ingenieurspraktikum beim ZFT Robotron. Dort wurden die aktuellen Publikationen zur relationalen Algebra gesichtet und somit der Wissenstransfer zwischen beiden Institution intensiviert, da beide jeweils über verschiedene Quellen-Bestände verfügten. Auch war zu diesen Zeitpunkt bei Robotron die relationale Algebra zwar bekannt, aber noch nicht weiterführend erforscht. Wie auch bei IBM war man dort in Befürworter und Zweifelnde gespalten. Die Befürworter hatten sehr positive Erwartungen, konnten jedoch kein funktionierendes Produkt vorweisen. Die Gegner hingegen wollten an den eingeführten und vor allem auch erfolgreichen Produkten, insbesondere dem am Netzwerkmodell orientierten DBS/R nicht rütteln. Demuth erinnert sich: »Man hat nicht geglaubt, dass relationale Datenbanken effizient sein können, weil man bei den relationalen Datenbanken mit Mengen arbeitet. Und es gibt keine Links von einem Datum zum anderen, man musste immer die Joins machen. Da erzeugt man natürlich viele Rechenoperationen, und bei den Netzwerk[modell]-Datenbanken konnte man über Referenzen schnell zum anderen Datum kommen« (Interview Demuth 2018, Fol 2). Zusammen mit den bereits erwähnten Problemorientierten Seminaren der TU Dresden, dürfte jedoch in diesem Zeitraum auch innerhalb des ZFT Robotron deutlicher geworden sein, dass man dem Thema

<sup>319</sup> Zum damaligen Zeitpunkt noch unter ihrem Geburtsnamen ›Starke‹, der mit der Heirat zu ›Demuth‹ wechselte.

nicht mehr aus dem Wege gehen konnte. Vorerst aber kehrte Birgit Demuth zum 8. Semester des Studiums zurück an die TU Dresden.

#### 3.3.4.2 Dissertationen

Nicht einmal zwei Jahre später, am 21.1.1980, reichten Bernd Keller und Michael Staruß ihre Dissertation *Entwurf und experimentelle Implementierung der COBOL-orientierten relationalen Datenmanipulationssprache REDAMS am Kleinrechner* ein.

Im politischen Diskurs der DDR ordnen die Autoren ihr Vorhaben mit einem Zitat des 9. Parteitag der SED im Mai 1976 ein, demnach als Ziel vorgegeben wurde, die »Erarbeitung naturwissenschaftlicher, mathematischer und technischer Grundlagen zur effektiven Beherrschung materieller und formalisierbarer geistiger Prozesse und ihrer zunehmenden Automatisierung in vielen Bereichen des gesellschaftlichen Lebens als entscheidende Voraussetzung für eine rationellere Gestaltung dieser Prozesse« (ebd., 5). Demnach situieren sie REDAMS als Grundlagenforschung zur Formalisierung materieller und geistiger Prozesse mit dem Ziel der Automatisierung. Ein weiterer wichtiger Punkt in diesem Zitat war das Interesse an Rationalisierung. Die Kleinrechner wurden in diesen Begründungszusammenhang aufgrund der »wesentlich geringeren Kosten« gestellt, denn das Ziel war es, »ein breites Spektrum von mittleren und kleineren Betrieben sowie Instituten mit Kleinrechnern auszustatten« (Keller/Staruß 1980, 6).

Konkreter wurde der Ansatz im Zuge Automatisierter Informationssysteme (AIS) positioniert, die eine Verwaltung der »Datenmengen« durch entsprechende »Datenbasissysteme« benötigten. In der Folge zählten die Autoren die Codd'schen Anforderungen an relationale Datenbankmanagement auf: Redundanzvermeidung, Mehrnutzerzugriff, Datenunabhängigkeit und Zugriffskontrolle. Es folgte als weitere Rahmung eine Liste unterschiedlichster nicht-relationaler und relationaler Datenbankmanagementsysteme aus der DDR und international (Abb. 73), wobei anhand von Literatur nachgewiesen wird, dass die »internationale Entwicklungstendenz« auf den Einsatz von Datenbankmanagementsystemen auf Kleinrechner verweise. (Keller/Starruß 1981, 1–3). Offensichtlich war immer noch strittig, ob die als ressourcenintensiv angesehenen relationalen Datenbankmanagementsysteme auf Kleinrechnern überhaupt lauffähig sein würden.

Die Autoren waren sich anhand der Analyse vorhandener relationaler Ansätze für Kleincomputer darüber im Klaren, vorerst nicht den vollen von Codd geforderten Funktionsumfang erreichen zu können. Als Grund dafür gaben sie vor allem die technische Beschränkungen im Speicherumfang der K 1600 Rechner an. Sie zeigten sich jedoch zuversichtlich, im Zuge der absehbaren technologischen Entwicklung eine vollständigere Software entwickeln zu können. Von SQL war zu diesem Zeitpunkt noch keine Rede, sie entschieden sich für eine eigene Datenmanipulationssprache auf Grundlage von COBOL, so wie es bereits in der Konzeption angelegt war. Dieser Abfragesprache REDAMS gingen sie zentral nach, indem sie einzelne Sprachelemente aufführten und diskutierten, wobei sie interessanterweise von Räumen anstatt von Mengen sprechen: Abstrakte Merkmalsräume, Objekträume, Bildung von Teilräumen und deren Projektion, Selektion und Vereinigung.

Kleinrechner Merkmale DBS	MONASH-MINI-DBMS	RISS	DBMS on the DGC HOVA	FEDMS	REDAMS (1. Ausbaustufe)
Typ der DML	DML mit Trägersprache FORTRAN IV	DML mit Trägersprache BASIC-PLUS und eigenständige DML mit Basisprache BASIC-PLUS	eigenständige DML mit Basisprache FORTRAN	DML mit Trägersprache COBOL, FORTRAN, u.a. und eigenständige DML	eigenständige DML mit Basisprache COBOL
Belegte HS-Kapazität	64 K Byte	unbekannt	unbekannt	ca. 28K Byte	48 K Byte
Nutzersichten (Datenmodelle)	Hierarchische und Netzwerkstrukturen	Menge von Relationen	Menge von Datensätzen	Hierarchische Strukturen, Menge von Datensätzen	Menge von Relationen
Ebene des Datenschutzes	Dateiebene	Relationenebene	unbekannt	Datenbankebene Dateiebene Datenelementebene	Relationenebene
Verwaltung verteilter Datenbasen	nein	nein	nein	ja	nein

Abb. 73: Einordnung von REDAMS im Verhältnis zu den diskutierten Datenbankmanagementsystemen. Das Akronym DML steht für Data Manipulation Language. HS-Kapazität meint die Hauptspeicher-Kapazität. Für REDAMS ist diese mit 48 Kbyte angegeben, so viel wie eine stark komprimierte JPG-Bilddatei von Passbildgröße, heutzutage (Keller/Starruß 1980, 17)

Dass die Dissertation Auswirkungen auf die weiteren relationalen Projekte der TU Dresden hatte, ist an einem konkreten Beispiel belegbar. In der Dissertation wurde erstmals von der Compilersprache CDL (Compiler Description Language) Gebrauch gemacht (ebd., 4). Diese übersetzte COBOL nach Assembler. CDL kam später auch in der Entwicklung des Compilers von DABA 1600 zum Einsatz und wurde im Nachgang von den Autoren im Vergleich zu den internationalen Entwicklungen als technisch herausragend gewertet (Demuth/Wiggert 2008, 282).

Zusammenfassend lässt sich sagen, dass die Autoren hier auf dem Stand ihrer Zeit argumentierten. Es handelt sich um eine pragmatische Aneignung der Argumentation und der technologischen Konzepte, die auf die vorhandene Kleinrechnertechnik der DDR angepasst wurden, wobei der eigentliche Zielrechner K 1600 noch nicht zur Verfügung stand und auf einem Großrechner des ESER-Systems EC-1022 getestet wurde. Entstanden war ein Prototyp, der überhaupt die Umsetzbarkeit der relationalen Algebra mit den in der DDR zur Verfügung stehenden Mitteln demonstrierte, nachdem bis dahin diese Möglichkeit vor allem theoretisch diskutiert wurde. In der Folge konnte der Studiengang an der TU Dresden auf den Erfahrungen aufbauen. Durch einen Hinweis in der Monatszeitschrift *rechentechnik/datenverarbeitung* wurde die DDR-Öffentlichkeit über den Prototypen REDAMS informiert (Keller/Starruß 1981).

Im Dezember 1981 veröffentlichte Vera Wenk an der TU Dresden ihre Dissertation *Untersuchungen zur effektiven Implementierung des Relationenkonzeptes für Datenbanksysteme am Beispiel relationen-algebraorientierter Anfragesprachen*. Im ersten Teil stellte sie den derzeitigen Stand der akademischen Diskussion ausführlich dar, in noch größerem Umfang als in den vorhergehenden Publikationen. Beispielsweise führte sie in einer Tabelle, in 13 Kategorien unterteilt, die verfügbare Literatur zur relationalen Algebra auf und kam dabei auf 95 Einträge (Abb. 74). Dies bedeutete nicht unbedingt, dass alle diese Artikel auch tatsächlich unmittelbar zugänglich waren. Vieles dürften Sekundärquellen gewesen sein, aber die Aufstellung gab einen sehr guten Überblick über das Forschungsfeld im damaligen

Zeitraum. Dieser Überblick entsprach Wenks Ziel, die relationale Algebra »umfassend« darzustellen und »einer Kritik zu unterziehen« (Wenk 1981, 15), wobei der historische und der aktuelle Stand miteinander verglichen werden sollten. Der Akt der Kategorisierung ist Zeugnis, wie sich diese Forschergruppe in das spezifische Themenfeld Relationales Algebra eingearbeitet hatte. Wenks Dissertation ist in der Folge auch häufig ausgeliehen und rezipiert worden, wie aus dem Zettel mit Rückgabefristen im Einband erkennbar wird (Abb. 75).

In einem weiteren Schritt untersuchte Wenk die verschiedenen Möglichkeiten der Anfrageoptimierung mit dem Ziel, »die gewonnenen Erkenntnisse für die Konzeption und experimentelle Implementierung einer relationenalgebraorientierten Anfragesprache anzuwenden und zu nutzen« (ebd.). Sie konstatierte, dass ein Großteil der Forschung an Universitäten und Forschungszentren stattfand und nur wenige Systeme bereits im produktiven Einsatz waren. Die Mehrzahl der bisherigen Prototypen erfolgte laut Wenk auf Großrechnern. Eine Tendenz zur Abfragesprache SEQUEL aufgrund ihrer Nutzerfreundlichkeit und der Möglichkeit diese in andere Sprachen (Host-Languages) einzubinden, sei erkennbar (ebd., 75f.).

Die Optimierungskomponente setzte Wenk in ein Verhältnis von Antwortzeit zu Speicherplatz. Eine direkte Übersetzung der englischsprachigen Anfrage nach PL/I würde aufgeblähten Code, oder wie Wenk schreibt »suboptimalen« Code erzeugen (ebd., 82f.). In diesem Bereich identifiziert sie ein Forschungsdesiderat, denn ein Großteil der ihr zugänglichen Optimierer beschrieb sie als angedockt an bereits existierende Software, während eine frühzeitige Integration in ein in Konzeption befindliches Datenbankmanagementsystem ausstünde. Ausgehend von der Veröffentlichung des IBM-Forschers Frank P. Palermo *A Data Base Search Problem* (Palermo 1974) entwickelte und verbesserte Wenk einen Optimierungsalgorithmus, wobei sie verschiedene Kategorien identifizierte, und wiederum mit mehreren möglichen Umformungsregeln verknüpfte und kritisch diskutierte.

Überblicksartikel zum Stand von Implementierungen o. mit einführendem Charakter	Normalformenlehre	Überführung von Datenmodellen	Schlüssel in Relationen	spezielle relationale Sprachen	unscharfe relationale Sprachen
BLASER u.a. 77 BRUDNO 76 BRUDNO 77 CHAMBERLIN 76 CHAMPINE 79 CODD 71 CODD 74 DATE 74 DATE 76 LUTZ 76 LUTZ 77 RAZMYSLOV 78 REYNOLDS u.a. 73 SCHEBER 76 SCHUBERT 77 SHNEIDERMAN 77 STONEBRAKER 77 WEDEKIND 76 WEDEKIND 78 ZALENKO 77	DELOBEL u.a. 73 CADIOU 76 DELOBEL u.a. 77 FAGIN 77 CHANG u.a. 80	MARTIN 75 FRASSON 75 KALINIČENKO 75a KALINIČENKO 75b KLUG u.a. 77	BÉKÉSSY u.a. 79a BÉKÉSSY u.a. 79b	ANTONACCI 77 ASTRAHAN u.a. 75a ASTRAHAN u.a. 76 BEZRUKOV u.a. 80 BOYCE u.a. 75 CODD 71 KELLER u.a. 80 NEUHOLD u.a. 77 STONEBRAKER u.a. 77 TODD 76 ZAMULIN u.a. 79 ZLOOF 75 ZLOOF 77	TAHANI 76 CHANG u.a. 78



spezielle relationale Datenbanksysteme	spezielle Probleme der Implementierung des Relationenkonzeptes	Nutzung des Relationenkonzeptes in Frage - Antwort - Systemen	Optimierung von Anfragen in relationalen Sprachen	spezielle Gerätetechnik für eine Implementierung des Relationenkonzeptes	psychologische Untersuchungen zur Nutzung relationaler Sprachen	relationales Interface in Systemen mit Mehrebenenarchitektur
ASTRAHAN u.a. 75b ASTRAHAN u.a. 76 CHAMBERLIN u.a. 76 FRASSON 75 HALL 76 HELD u.a. 75 MCLEOD 75 NEUHOLD u.a. 75 SCHEBER 77 SCHMID 76 STONEBRAKER 76 SYSTEM_R GROU P 79 TODD 76 TSICHRITZIS 75 WHITNEY 74 ZLOOF u.a. 77	ASTRAHAN u.a. 80 BECK 80 BLASGEN u.a. 77 CHAMBERLIN u.a. 75 HÄRDER 75a HÄRDER 76 HÄRDER 77 HÄRDER 78 LORIE u.a. 79	BELKE u.a. 80 SAGALOWICZ 77 WALTZ u.a. 76 WALTZ u.a. 78 WERNER 79a WERNER 79b	AHO u.a. 79 ASTRAHAN u.a. 75 CODD 71 HALL 76 HÄRDER 78 PALERMO 74 ROTHNIE 75 SCHMID 76 SMITH u.a. 75	BAUM u.a. 76 CHAMBERLIN 79 OZKARAHAN u.a. 75 SMITH u.a. 79	THOMAS u.a. 75 REISNER u.a. 75	KALINIČENKO 75a KALINIČENKO 75b KLUG u.a. 77 MYLOPOULUS u.a. 75

Abb. 74: Vera Wenks Matrix von Veröffentlichungen zur relationalen Logik (Wenk 1981, 10f.).

Buch Nr.		Dieses Buch ist zurückzugeben bis zum	
- 8 10 82	14.08.84		
03.12.82			
13. Feb. 1983	15. Okt. 1984		
04.03.83	14. Dez. 1984		
12.04.83	1.03.85		
26.04.83	18. März 1985		
1. Juli 1983	5.1. März 1986		
5. Aug 1983	16.05.86		
-7.10.83	25. Nov. 1987		
6. Nov 1983	5. Mai 1987		
08. Jan 1984	13. Juni 1987		
5. März 1984	10. Juli 1987		
9. März 1984	26. Juli 1989		
16. April 1984	18.10.89		
17. Mai 1984			

Bestell-Nr. 0520 (204) LG 39/206/77 1000 1072

Abb. 75: Ausleihzettel der Dissertation Wenk, Universitätsbibliothek Dresden (Wenk 1981).



Abb. 76: ESER EC-1022 Großcomputer 1982 mit Tastaturterminal zur Eingabe, Magnetbandschränken zur Speicherung und Bedienkonsole im Bildhintergrund im Zentrum der staatlichen Archivierung der Raumfahrtokumentation Moskau. Dieser in der UdSSR und in Bulgarien produzierte Großrechner war Teil des ESER-Systems und stand auch für die Studierenden der TU Dresden zur Verfügung (Центр государственного хранения космической документации 1982).



Abb. 77: Dateneingabe am ESER EC-1022 Großcomputer, 1982. Die auf den tabellarisch angelegten Vordrucken per Hand geschriebenen Programmieranweisungen werden von den Operatorinnen mithilfe einer spezifischen Tastatur auf Lochkarten übertragen, welche in der Folge am Großrechner als Programm eingelesen werden können (Центр государственного хранения космической документации 1982).

Im dritten Teil ihrer Dissertation konzipierte sie eine an relationaler Algebra orientierte Abfragesprache für den Großrechner ESER EC 1022 (Abb. 76, Abb. 77). Ziel war es, eine relational vollständige Abfragesprache FQS zu konzipieren und die günstigste Optimierungskomponente nach PL/I zu finden (ebd., S.159ff.).

Wenk lieferte mit Überblick und Klassifizierung von Optimierungsansätzen eine wichtige Vorarbeit, auf die spätere Projekte aufsetzen konnten. Sie evaluierte einen Großteil der damals bekannten Ansätze zur Abfrage-Optimierung – ein Wissen, welches in das Folgeprojekt DABA 1600 einfließen konnte, an dem sie unmittelbar als Projektleiterin beteiligt war. Der Ansatz, eine relationale Sprache für ESER-Großrechner zu schaffen, wurde von der Forschungsgruppe nicht weiter verfolgt. Stattdessen orientierte sie auf Kleinrechner um. Das mag daran gelegen haben, dass mit RODAN SQL vom Centrum Projektovenia i Zastosowan Informatyki in Warschau und einer SEQUEL 2 Implementation des MUSON (Zentrales ökonomisch-mathematisches Institut der Akademie der Wissenschaften der UdSSR) bereits zwei ESER-kompatible SQL-Projekte für Großrechner in Entwicklung waren (Starke (Demuth) 1983, 14). Wichtiger noch war sicherlich geeignete Software ›westlichen‹ Ursprungs die durch Robotron geplant wurde, zum Beispiel das in ALLDBS umbenannte Oracle. Ebenfalls eine Rolle spielte die Ausrichtung der Vertragsforschung auf Kleinrechner. Diese Orientierung kam Robotron entgegen, da es bereits mit DBS/R über ein erfolgreich eingesetztes, nicht-relationales Datenbanksysteme auf Großrechnern verfügte und dies als ausreichend ansah. Nachdem der Wissenstransfer bis hierher hauptsächlich theoretischer Natur war, bewegte er sich nun in das Feld der praktischen Umsetzung.

#### 3.3.4.3 Realisierung

Birgit Demuths, im September 1983 erschienene Dissertation *Übersetzung und Optimierung relationaler Anfragen am Beispiel der Datenbanksprache SQL* besaß im Unterschied zu den bisher besprochenen Forschungsarbeiten einen anderen Status. Es handelte sich nicht mehr allein um eine Vorarbeit, sondern um eine zentrale Ausarbeitung, aus der heraus die anwendbare Abfragesprache SQL 1630 für DABA 1600 entstand. Sie ist ins Verhältnis zu setzen zum Beginn der Vertragsforschung von ZFT Robotron und TU Dresden 1980, nachdem die Vorlaufarbeiten seit 1977 im Gange waren (Schubert in: Interview Demuth 2018, Fol 2). Demuth war seit 1980 Forschungsstudentin in einer Arbeitsgruppe, die von Vera Wenk geleitet wurde und zu der noch weitere wechselnde Assistenten und Studierende mit Diplomarbeiten gehörten.<sup>320</sup>

Über Quellcode von INGRES oder System R verfügte die Autorin und ihre Kollegen nicht: »Wir haben uns nur die Prinzipien, worauf man achten muss, angeschaut und die Artikel worüber theoretisch geschrieben worden ist, haben wir versucht zu verstehen. [...] Wir haben die Ansätze, die es gab, nachvollzogen. Die waren ja auch nicht im Detail beschrieben, [und dann haben wir] die Ideen aufgegriffen und das selber weiterentwickelt« (Demuth 2018, Fol 6). Eine der Schwierigkeiten in der Forschung ergab sich aus einer Ungleichzeitigkeit: Da das ZFT Robotron mit dem unterliegenden Datenbankbetriebssystem, auf welches das SQL der TU Dresden aufbauen sollte, langsamer voran kam, konnten Demuth und ihre Kollegen die Leistungsfähigkeit nur theoretisch ermitteln. Sie entwickelten mit Hinblick auf in Gesprächen vereinbarte Schnittstellen, die später jedoch einer Revision unterzogen werden mussten, nachdem die Arbeiten bei Robotron voranschritten. Praktische Tests der Optimierungskomponente waren während der Dissertation nicht möglich und erfolgten wohl erst nachdem das

---

<sup>320</sup> Zum Beispiel: Möser: *Beiträge zur Verbesserung und Erweiterung des 1. Passes des SOL 1630-Compilers DABA 1600*. Diplomarbeit 1985; Hermannsdörfer: *Beiträge zur Implementierung von DABA 1600*. Diplomarbeit 1985 (zitiert n. Schubert 1985, 112).

Forschungsvorhaben an der TU Dresden abgeschlossen und an Robotron zur Entwicklung übergeben war (ebd., Fol 7). Die programmiertechnische Umsetzung verortete sich im Robotron Rechenzentrum, das zentral in der Stadt auf dem Robotron-Campus Leningrader Straße (Heute St.-Petersburger-Str, in der Nähe des Pirnaischen Platzes) gelegen war, denn dieses verfügte frühzeitig über K 1600 Rechner. Die Programme entwarf Demuth zuerst auf Papier, später konnten sie auch direkt im Rechenzentrum eingegeben werden. Die Sicherung der Arbeiten erfolgte auf externen Speicherplatten (EC 5269) mit einem Speicherumfang von 2,5 Mb, welche zwischen dem Rechenzentrum und der Universität jeweils hin- und her zu tragen waren (ebd., Fol 2). Die Kassette maß im Durchmesser 50 cm – ein beschwerliches Ungetüm. Ihre Arbeitsbedingungen stehen dabei paradigmatisch für die Forschergruppe. Die Dissertation war den bereits diskutierten Veröffentlichungen recht ähnlich gerahmt. In der Einleitung verwies die Autorin auf die wachsende Verbreitung von Kleinrechnern in der DDR: »Eines der wichtigsten Ziele dieses Rechnereinsatzes ist die Unterstützung geistiger Prozesse in Forschung und Entwicklung, in der Projektierung, Konstruktion und technologischen Vorbereitung der Produktion« (Starke (Demuth) 1983, 1). Daraus ergäben sich »qualitativ neue Anforderungen«, da die verbreiterte Nutzerbasis auch eine Zugänglichkeit für »Nichtprogrammierer« bedeute. Es wurde auch mit dem »internationalen Maßstab« argumentiert, zum Beispiel mit der Situation in den USA: »So sind z. B. im Jahre 1980 von den 54 in den USA kommerziell verfügbaren DBBS [Datenbankbetriebssystemen], 19 DBBS für Klein- und Mikrorechner; davon wiederum 10 DBBS für Rechner der Familie PDP-11 unter Steuerung des Betriebssystems RSX-11M; [und] 4 relationale Kleinrechner-DBBS [...]« (Starke (Demuth) 1983, 3f.). Daran anschließend diskutiert die Studie die Vor- und Nachteile der relationalen Algebra und schätzt ein, dass die Auseinandersetzungen um relationale Datenbanksysteme grundsätzlich »der Entwicklung der Datenbanktechnologie zu einem bedeutenden Aufschwung verholfen haben« (ebd., 6). Sie kennzeichnete damit den vorangegangenen Streit als produktiv. Im Unterschied zu den vorrangig konzeptuellen Arbeiten Ende der 1970er Jahre konstatiert sie für die beginnenden 1980er die zunehmende Überführung von Datenbanksystemen aus dem Forschungsstadium in Softwareanwendungen und hebt die Datenbanken INGRES (University of Berkeley) und SQL/DS (IBM) als Beispiele heraus. Für die weitere Beforschung von SQL spräche dessen zunehmende internationale Bedeutung.

In der Zusammenfassung ihrer Dissertation findet sich ein bemerkenswerter Satz: »Der Einsatz der Datenbanktechnologie im Rahmen automatisierter Informationssysteme wird daher mehr und mehr zum objektiven Erfordernis und zu einer wichtigen Voraussetzung für die Gewährleistung des notwendigen Leistungsanstieges in der Volkswirtschaft« (ebd., 144). Wann und wie Datenbanken zu einem »objektiven Erfordernis« in der DDR wurden, ist eine der Fragestellungen, denen dieses Kapitel nachgehen sollte. Sicher ist er in die eingangs erwähnte Neuausrichtung der DDR auf die »unmittelbare Produktivkraft Wissenschaft« einzuordnen. Demuths Dissertation belegt, dass die Einführung der relationalen Algebra erfolgte, als durch Netzwerk- und Hierarchische Datenbanksysteme die Verwendung von Datenbanken bereits grundlegend in die Infrastrukturen der Informationsverarbeitung eingewoben war. Die industriellen und organisatorischen Prozesse hatten auch in der DDR eine Komplexität angenommen, welche nur mittels Datenbanktechnologie »objektiv« bewältigbar blieb.

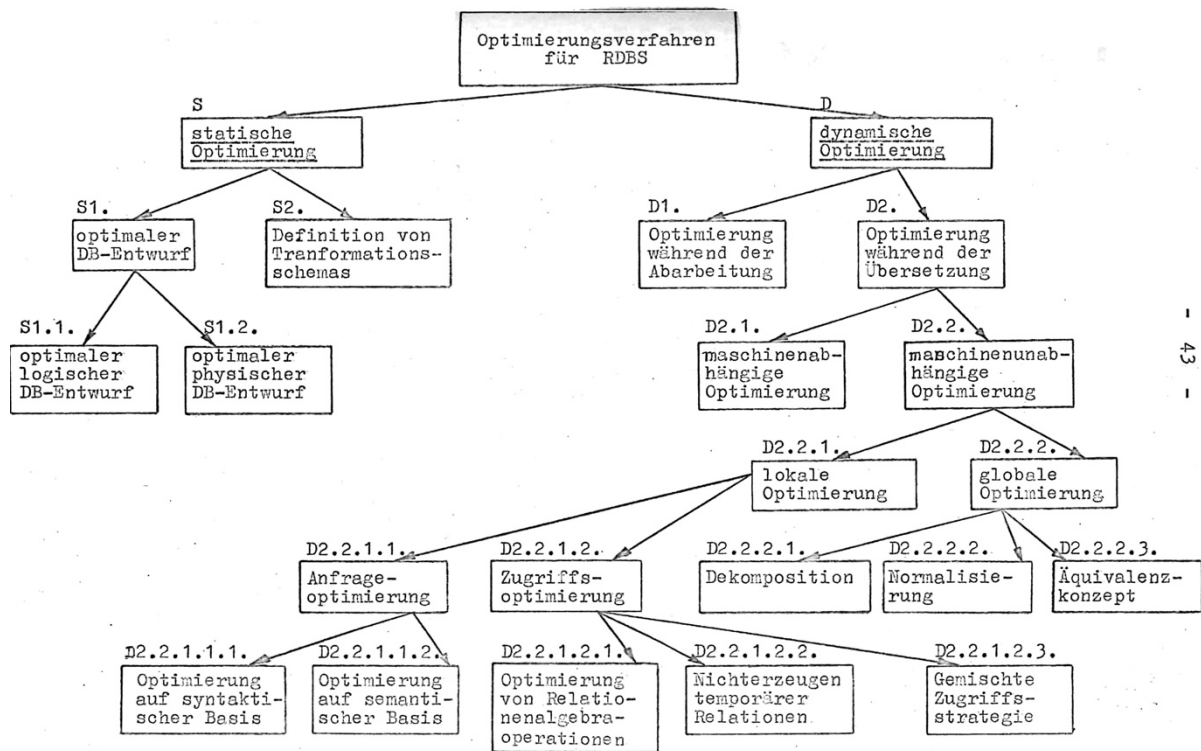


Bild 3.1. Optimierungsverfahren für RDBS

Abb. 78: Verschiedene Optimierungsstrategien, die in der Dissertation diskutiert werden. (Starke (Demuth) 1983, 43).

Aufbauend auf der vorhergehenden Forschung ihrer Forschungsgruppe diskutierte das zweite Kapitel von Demuths Dissertation den damals aktuellen Sprachstand von SQL und das dritte und vierte Kapitel untersuchten den aktuellen »Weltstand« der Optimierung relationaler Sprachen mit dem konkreten Ziel, den SQL Compiler zu konzipieren.<sup>321</sup> Dann wurden anhand der verfügbaren Forschungsliteratur verschiedene Optimierungsstrategien für relationale Datenbankmanagementsysteme systematisiert (Abb. 78) und damit die Voraussetzung für einen eigenen, dem Kleinrechner K 1600 angemessenen Ansatz geschaffen.

Als nächstes wandte sich Demuth dem Mehrschichtenmodell zu. Bereits seit einiger Zeit war in den USA das sogenannte ANSI-SPARC-Mehrschichtenmodell diskutiert worden. Die Computer-Figuration wurde in diesem Konzept in mehrere Schichten unterteilt. Die oberste, »externe«, Ebene betraf die Benutzeroberfläche, das heißt alle Elemente, mit denen die Nutzer direkt interagieren konnten. Dazu zählten neben dem graphischen User-Interface auch die Abfragesprachen, wie zum Beispiel SQL. Analog zu dieser oberen Schicht sprach man von einer unteren, »internen« Schicht, welche für die Speicherung von Daten auf dem Datenträger zuständig war. Sie organisierte automatisch, an welcher Adresse im Speicher Daten abgelegt wurden und wie diese automatisch auffindbar waren. Zwischen beiden vermittelte eine weitere, sogenannte »konzeptionelle« Ebene. Diese konzeptionelle Ebene konnte dem Netzwerk- oder hierarchischen Modell folgen, oder eben der relationalen Algebra (Abb. 79).

<sup>321</sup> In der vorliegenden Studie wird dieses Thema nur knapp gestreift. Für eine eingehende Diskussion der ANSI-SPARC-Mehrebenenarchitektur siehe *Digitale Datenbanken – Eine Medientheorie im Zeitalter von Big Data* (Burkhardt 2015, 221–242).

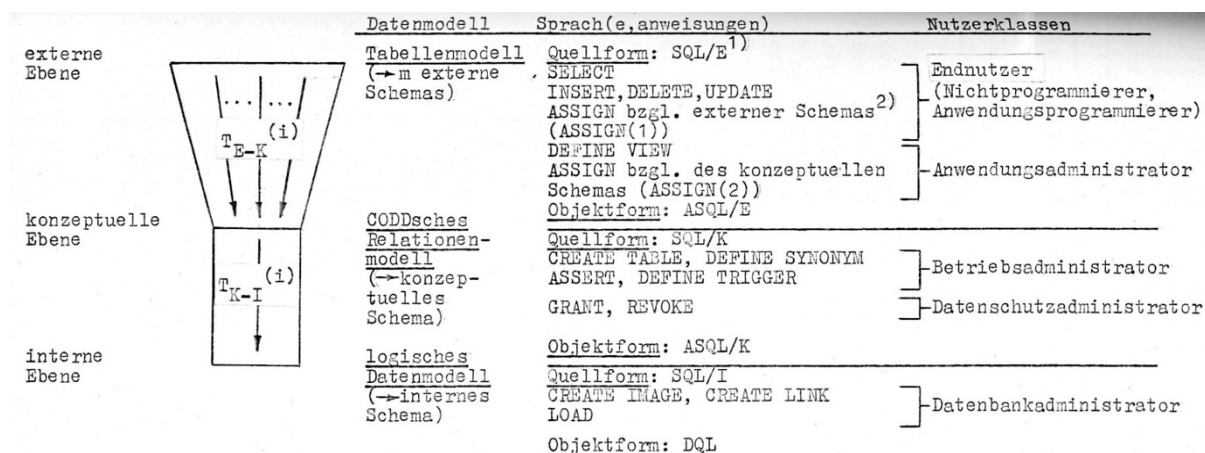


Abb. 79: Schichtenmodell einer relationalen Datenlogik. Der Begriff Datenschutz stand in der DDR für die Zugriffsrechte, nicht für den Schutz der Privatsphäre. Somit war ein Datenschutzadministrator für die Verteilung von Zugriffsprivilegien (wer darf was lesen oder verändern) zuständig. Die Ausdifferenzierung der Nutzerklassen ist heute im Wesentlichen auf die Anwendungsprogrammierer und Datenbankadministratoren geschrumpft, da nicht – wie damals angenommen wurde – Abfragen durch Endnutzer per SQL direkt erfolgen, sondern diese für gewöhnlich in Softwareanwendungen eingebunden sind (Starke (Demuth) 1983, 70).

Die Ebenenbeschreibung demonstrierte konkret, welche Operationen Menschen mit relationalen Datenbanksystemen ausführen können und gab eine Art Fahrplan vor, wie die dafür nötige Infrastruktur idealerweise aufgebaut sein konnte. Daraus ergaben sich auch Hinweise für die Entwicklungsschritte, die notwendig waren, um vom Konzept zum Produkt zu gelangen.

Abschließend verglich Demuth ihren eigenen Lösungsvorschlag mit denen von System R und INGRES. Durch eine Reihe von Erweiterungen erreichte die Autorin, nach eigener Einschätzung, einen größeren »Freiheitsgrad der Optimierung« gegenüber System R (Starke (Demuth) 1983, 135). Dabei war in der erreichten Funktionalität der Dresdener SQL 1630-Compiler an den kalifornischen System R-Compiler prinzipiell angelehnt, während sie sich im internen Aufbau voneinander unterschieden (ebd. 137).

### 3.3.5 Übergang von der Forschung und Entwicklung in die Produktion

Der Übergang vom Forschungsprojekt zum Produkt verlief nicht reibungslos. Bereits angedeutet wurde die Ungleichzeitigkeit der Entwicklung. Beim ZFT Robotron war der Bereich »Anwendung« ab 1979 mit DABA 1600 befasst, zwei Jahre später wurde die Gruppe betriebsintern noch einmal umgestellt und stand fortan unter der Leitung von Karl-Heinz Wiggert. »Der Alltag der Entwickler war, gemessen an heutigen Arbeitsbedingungen, sehr mühsam« (Demuth/Wiggert 2008, 285). Dies betraf nicht allein den teils schwierigen Zugriff auf Rechenkapazitäten der Großcomputer im Rechenzentrum, welche oft nur in Randzeiten und nachts verfügbar waren, sondern auch zusätzliche zweckferne Arbeitseinsätze, z.B. zur Schneeberäumung der Straßen: »Da fanden sich dann hochqualifizierte diplomierte und promovierte Fachkräfte statt am Rechner, am Schreibtisch oder im Labor, mit der Schneeschippe auf der Straße wieder« (ebd.).

Unabhängig davon intensivierte Robotron ab 1981 sein Engagement, allerdings erst nachdem der SQL-1630 Compiler und Optimierer seitens der TU Dresden entwickelt war. So begann Robotron ernsthaft, personelle Kapazitäten für die Entwicklung der unterliegenden Schicht, vor allem des Datenbankbetriebssystem, zur Verfügung zu stellen. Retrospektiv schätzte Demuth die Situation aus Perspektive der Forschung als problematisch ein: »Also wir haben dort ganz schön Federn gelassen und gemerkt, dass

man als Universität sich nicht an einer kommerziellen Implementierung beteiligen soll. Wir müssen Prototypen liefern, wir müssen Forschung machen« (Interview Demuth 2018, Fol 11). Dies deckt sich mit der Einschätzung des Historikers Reiner Pommerin, demnach die zunehmende und teils überwiegende Auftragsforschung zu Lasten der Grundlagenforschung an der Universität ging. Begründet war dies in der industrieorientierten Bildungskonzeption der 1960er Jahre, im Zuge der Neuausrichtung der Planwirtschaft durch das *Neue Ökonomische System* (Pommerin (Hrsg.) 2003, 1:296).

Nach Ende des Forschungsprojektes 1983 orientierte die Sektion Informationsverarbeitung die Forschung in Richtung CAD/CAM (Computer Aided Design/Computer Aided Manufacturing), dem großen Staatsplanthema jener Jahre. Während viele ihrer Kommilitonen die Universität mit Ende des Studiums verlassen hatten, blieb Demuth an der TU Dresden und damit auch dem Projekt DABA 1600 verbunden. Bis November 1984, dem Ende der Industrieprobung und Beginn des offiziellen Vertriebs als Produkt, war sie an der Vervollständigung des Compilers in Zusammenarbeit mit dem ZFT Robotron beteiligt und danach für Supportaufgaben der bereits existierenden Anwender zuständig. Erst 1986 gingen diese Aufgaben endgültig von der TU Dresden an Robotron über. »Während die Forschungskoooperation zu wissenschaftlichen Einrichtungen aus ZFT-Sicht generell als wesentliche Verstärkung der kombinatseigenen Forschungskapazitäten eingeschätzt wurde [Jor06], fühlten sich die wissenschaftlichen Mitarbeiter oftmals als reine Programmierer mißbraucht. [...] Es konnte jedoch trotz des Wunsches nach Technologietransfer nicht Ziel einer Universität sein, als verlängerter Arm eines Softwarehauses umfangreiche Programmierarbeiten zu leisten und damit die wissenschaftliche Arbeit zu vernachlässigen«, lautet die selbstkritische Einschätzung einiger Beteiligter mit zweiundzwanzig Jahren Abstand (Demuth/Wiggert 2008, 289f.).

Es fügte es sich, dass in den vorhergehenden Jahren eine Reihe ehemaliger Studierender der TU Dresden zu Robotron übernommen worden. Von diesem Team wurde die umfangreiche Dokumentation erstellt, die Software auf Magnetbändern standardmäßig ausgeliefert, sowie Kundensupport geleistet (Demuth/Wiggert 2008, 287f.; Interview Demuth 2018, Fol 11). Zu den Erstanwendern zählten unter anderen der VEB Chemieanlagenbau Germania, Chemnitz, der VEB Waggonbau Ammendorf/Halle(Saale) und Dessau, der Kühlgerätebau VEB DKK Scharfenstein (Erzgebirge) und das Moskwitsch-Automobilwerk, Moskau.

### 3.3.6 Anwenderinformationen

Zum Übergang aus Forschung und Entwicklung in die Produktion, zur Werdung eines Software-Produktes, gehört, neben den umfangreichen Praxistests und Testimplementationen, die Erstellung einer Anwenderdokumentation. Der generellen *Anwendungsbeschreibung* (VEB Robotron 1981), noch in der Phase der Produktentwicklung, folgten drei Jahre später *Methodische Hinweise* (VEB Robotron 1984a) und eine *Sprachbeschreibung* von SQL-1630 (VEB Robotron 1984b), die den Übergang in den produktiven Einsatz von DABA 1600 kennzeichnen. Während die ersten beiden Publikationen durch »Ein Kollektiv des ZFT Robotron« erstellt wurden, stammt die Sprachbeschreibung vom »Kollektiv der Sektion Informationsverarbeitung der Technischen Universität Dresden«. Diese aktive Vermittlung wurde nach Einschätzung der Robotron-Autoren Franz Menhart und Jürgen Stumpf nötig, da »die Anwendung der Datenbanktechnologie bei Kleinrechnern noch relativ neu ist – die Situation unterscheidet sich hier völlig von der bei Großrechnern [...]« (Mehart/Stumpf 1987a, 18). Die dreiteilige Produktdokumentation soll als diskursives Dokument ernst genommen werden und einer Reihe von Fragen

ausgesetzt werden: Welche Argumentationen spielten eine Rolle? Entfalteten sich Widersprüche? Welches waren die Stärken und Schwächen aus Sicht seiner Entwickler?

Die erste von drei Dokumentationen, 1981 vom ZFT Robotron, eröffnete mit dem Thema der Informations-Management-Systeme,<sup>322</sup> die in der DDR als Automatisierte Informationssysteme (AIS) bezeichnet wurden. Schwerpunkt der Argumentation war das Anwachsen der Datenmassen und deren ständige Aktualisierung, sowie der Abschied von der sequentiellen, Maschinen-orientierten Stapelverarbeitung, in Richtung einer Datenverwaltung, welche sich an abstrahierten Logiken orientiert. Die Dokumentation orientierte auf »die Bereitstellung geeigneter sprachlicher Mittel für die Mensch-Rechner-Kommunikation, die der natürlichen bzw. Fachsprache des Nutzers ähnlich sind und nur geringen Einarbeitungsaufwand erfordern« (VEB Robotron 1981, 11) und verwies damit auf die, zu diesem Zeitpunkt in Entwicklung befindliche, Sprache SQL 1630. Als Vorteil beschrieben die Autoren den Umstand, dass durch die Nähe zur natürlichen Sprache nicht nur Systemadministratoren, sondern auch einfache Nutzer auf die Datenbestände zugreifen könnten und zum Beispiel Abfragen in Auskunfts- und Buchungssoftware auch durch Nichtprogrammierer bewältigt werden könnten. Als Sprache brachten sie SEQEL 2 (welches später in SQL umbenannt wurde) ins Spiel, welches ursprünglich durch IBM im Zuge von System R entwickelt wurde und für DABA 1600 in leicht verkleinerten Funktionsumfang programmiert wurde.

»Der Anwendungsbereich von DABA 1600 erstreckt sich auf alle Sachgebiete und Problemklassen, bei denen die relevanten Informationsbestände in die geforderte Tabellenform überführt werden können oder diese möglichst sogar von Natur aus besitzen. Diese Tabellenform ist beispielsweise von vornherein gegeben bei Informationen über Personen, über Gesamtheiten von Arbeitsgegenständen, Arbeitsmitteln, Waren und ähnlichen Objekten, über Ergebnisse von Experimenten bei Forschungsarbeiten u.a.m.« (ebd., 16). Den potentiellen Nutzern lieferten die Autoren damit einleitend vor allem Argumentationsmaterial. Schließlich ging es darum, einerseits überhaupt eine zentrale Datenhaltung in Datenbanken durchzusetzen und andererseits, mit Hilfe der relationalen Algebra, ein Plädoyer für die maschinenunabhängige und datenunabhängige Speicherung zu formulieren. Nach dieser Einleitung beschrieben sie die konkreten Techniken, Vorgehensweisen und Sprachelemente und erläuterten ein Anwendungsbeispiel.

Bereits ein Jahr zuvor, 1980, wurde der Kleinrechner Robotron K 1600 in der Zeitschrift *Neue Technik im Büro* erstmals der DDR-Öffentlichkeit angekündigt (Abb. 80). Der Autor Horst Giebler, einer der Konstrukteure, stellte einen modularen Mikrorechner in Aussicht, der in das System Kleinrechner (SKR) eingegliedert war und mit Komponenten anderer Hersteller aus dem RGW kombinierbar war (Giebler 1980, 39–40). Im Folgejahr erschien eine erste Darstellung von DABA 1600, welche es als modulares Mehrnutzersystem nach dem 3-Schichtenmodell ankündigt und der »Trennung der Daten von der Programmorganisation« genügt (Horn/Stumpf/Ziegenbalg 1981, 1–2). Dem Ziel der Popularisierung der relationalen Algebra anhand von DABA 1600 diente auch ein längerer Artikel in *Neue Technik im Büro* von 1982, welcher als Beispielanwendungen Auskunfts- und Buchungssoftware und allgemeine Anwendungsprogramme positionierte (Wiggert/Bulla/Ernst 1982).

---

<sup>322</sup> Zu den Informationssystemen des Westens siehe *Inventing Information Systems – The Systems Men and the Computer, 1950–1968* (Haigh 2001).



Tafel 1. Bedien-, Datenverarbeitungs- und Prozeßperipherie des Rechnersystems robotron K 1600

Nr.	Gerätebezeichnung	Kurzbezeichnung	Technische Daten	Geräteinterface	Zugeordnete AS	Rechnerseitiges Interface
1	Lochkartenleser	VT 42111/UVR CM 6101	600 Karten/min Auf Tischgerät	IFSP	ASP	Systembus K 1600
2	Lochkartenleser	K 6102/DDR CM 6102	160 Karten/min Auf Tischgerät	IFSP	ASP	Systembus K 1600
3	Lochbandeinheit	K 6200/DDR —	LBL: 200 Zeichen/s LBS: 50 Zeichen/s Einschub	IFSP	ASP	Systembus K 1600
4	Lochbandleser	robotron 1210/DDR —	200 Zeichen Auf Tischgerät	SIF 1000	ASI	Systembus K 1600
5	Lochbandstanzer	robotron 1215/DDR —	50 Zeichen Auf Tischgerät	SIF 1000	ASI	Systembus K 1600
6	Lochbandeinheit	CM 6204/VRP	LBL: 1000 Zeichen/s LBS: 50 Zeichen/s Einschub	IFSP	ASP	Systembus K 1600
7	Lochbandleser	FS 1503/CSSR	1500 Zeichen/s Auf Tischgerät	IFSP	ASP	Systembus K 1600
8	Kassettenmagnetbandeinheit	K 5261/DDR	2 × Kassettenmagnetbandgeräte/Einschub	IFSS	ASD	Systembus K 1600
9	Seriendrukker	robotron 1152/DDR	30 Zeichen/s bei 132 Zeichen/Zeile Auf Tischgerät Standgerät	IFSS	ASD	Systembus K 1600
10	Paralldrucker	VT 27060/UVR —	600 Zeilen/min 64 Zeichen/Zeile Standgerät	IFSP	ASP	Systembus K 1600
11	Bedieneinheit	K 7160/DDR —	31 cm Bildschirm 24 Zeilen mit 80 Zeichen alphanumerische Kompakt- tastatur Auf Tischgerät	IFSS	ASD	Systembus K 1600
12	Fernschreibmaschine	FSM T 91, T 63, DDR —	50 Bit/s Auf Tischgerät	IFS 1	ASD	Systembus K 1600
13	Bildschirm-Eingabe-/Ausgabegerät	K 7260/DDR —	31 cm Bildschirm 24 Zeilen mit 80 Zeichen alphanumerische Kompakt- tastatur Auf Tischgerät	IFS 2	ASD	Systembus K 1600
14	Prozeß-Eingabe-/Ausgabeeinrichtung	Ursadat 5010/DDR —	Baugruppensortiment Analog-Ein-/Ausgabe, Digital-Ein-/Ausgabe, Impulseingabe, Inkrementalausgabe usw. Schrank	ASU-TP	ATP	Systembus K 1600
15	Magnetbandgerät	ISOT 5004/E/VRB CM 5300	10 KByte/s 32 Bit/mm Einschub	IFMB	AMB	SKR-Einheitsbus
16	Magnetbandgerät	ISOT 5006 CM 5303	36 KByte/s 32 Bit/mm Einschub	IFMB	AMB CM 5001	SKR-Einheitsbus
17	Kassettenplatten-speicher	ISOT 1370/VRB CM 5400	1 Festplatte 1 Kassettenplatte mit je 25 MBit, 2,5 MHz, 60 ms, Einschub	IFKP	AKP	Systembus K 1600
18	Kassettenplatten-speicher	MERA 9425/VRP CM 5401	1 Festplatte 1 Kassettenplatte mit je 25 MBit, 2,5 MHz, 60 ms, Einschub oder Standgerät	IFKP	AKP	Systembus K 1600
19	Follenspeichereinheit	UVR/DDR	2 × MF 3200 Einschub	GNI	AFS	Systembus K 1600

Abb. 80: Diese Übersicht erschien in der zweimonatlichen Zeitschrift Neue Technik im Büro: K 1600 als Beispiel für das System der Kleinrechner (SKR) der RGW-Staaten. Alle Komponenten, die nicht mit 'robotron' gekennzeichnet sind, kommen aus den anderen Teilnahmeländern des SKR. Das ist der Planungsstand, der keine Aussage über die tatsächliche Verfügbarkeit darstellt (Giebler 1980, 39–40).

Zwischen der ersten Robotron-Anwenderdokumentation und den weiteren Veröffentlichungen vergingen zweieinhalb Jahre, in denen DABA 1600 Produktreife erlangte. Im November 1984 veröffentlichte Robotron die bereits erwähnten *Methodischen Hinweise* und die *Sprachbeschreibung* von SQL 1630. Diese Publikationen nahmen weitere Erfahrungen aus den Anwendungsprototypen auf. So dienen die *Methodischen Hinweise* als Orientierung, da die betreuenden Mitarbeiter von Robotron festgestellt hatten, dass die Einstiegserfahrungen der Nutzer sehr heterogen waren. Sie geben jedoch einen großartigen Einblick in die Bedürfnisse der Anwender (aus Perspektive von Robotron): »Typische Beweggründe des Anwenders zur Entscheidung für die DB können z.B. sein:

- Erwiesene Mehrfachnutzbarkeit großer Datenbestände durch unterschiedliche Nutzer bzw. Programme (damit Einsparung an Speicherplatz durch nur einmalige Speicherung dieser Daten)
- Zweckmäßige Trennung zwischen Anwenderprogrammen und den zu verarbeitenden Daten wegen:
  - häufiger Änderungen in den mehrfach nutzbaren Datenbeständen,
  - häufiger Änderungen in den Anwenderprogrammen ohne Auswirkung auf die von ihnen verwendeten Daten (damit Einsparung an Arbeitszeit in Aktualisierungsetappen)«(VEB Robotron 1984a, 8).

Diese Beweggründe entsprechen im Großen und Ganzen den Motivationen, die auch die westlichen Entwickler als Anlässe für die Erforschung neuer Datenbanklogiken angaben. Es ist bezeichnend, dass sich die Anwendungsbedürfnisse in den sozialistischen und kapitalistischen Ländern sehr ähnelten und nicht größere Unterschiede zeigten.

Eines der Probleme von Robotron scheint gewesen zu sein, dass die an DABA 1600 auf einem Kleinrechner heran getragenen Erwartungen sehr hoch waren und der Performance von DBS/R auf einem Großrechner entsprachen. So warnten die Autoren, die Leistungsgrenzen eines Kleinrechners unbedingt im Blick zu behalten und forderten dazu auf, eine Datenbank »auf dem hier erreichbaren Niveau« zu planen (VEB Robotron 1984a, 8). Dazu gehöre auch, eine gewisse Erweiterbarkeit bereits im Voraus einzuplanen, da sobald eine Softwarefiguration im produktiven Einsatz sei, sich weitere Wünsche der Nutzer ergäben. Diese Diskussion verweist auf die allgemein konstatierte Ressourcenknappheit an Rechentechnik jener Zeit, denn der Kleinrechner K 1600 ist auch eine Reaktion auf den Umstand, dass der Erwerb und Betrieb von Großrechnern für kleinere organisatorische Einheiten schlichtweg zu teuer war.

Die dritte Dokumentation von DABA 1600 – die *Sprachbeschreibung* – diente als eine möglichst breite Nutzerschichten ansprechende Darstellung der Datenmanipulationssprache SQL 1630, die einen Großteil des internationalen SQL-2 Standards abbildete. Eingangs beleuchteten die Autoren noch einmal den internationale Kontext, beispielsweise, dass SQL von IBM im Zuge von System R entwickelt wurde. Im Unterschied zu den Adaptionen Robotron ALLDBS (Oracle) oder Robotron Redabas (dBase II) tritt hier ein berechtigtes Selbstbewusstsein über das selbst Geschaffene zu Tage.

Immer wieder wurde auf die einfache Nutzbarkeit verwiesen: »Gegenüber den gewissermaßen als mathematischen Modellen relationaler Anfragesprachen dienenden Sprachen der Relationenalgebra und des Relationenkalküls zeichnet sich SEQUEL 2 durch eine erhöhte Nutzerfreundlichkeit insbesondere gegenüber Nutzern ohne rechen- und programmtechnische Spezialkenntnisse aus« (VEB Robotron 1984b, 8). Ungeübte Anwender sollten sich innerhalb von zwei Stunden einfachste Sprachkenntnisse aneignen können.

Ungewöhnlich für die DDR war, dass nach einer langen Phase der Eindeutschung englischer Begriffe den Anwendern zugetraut wurde, die Software mittels englischsprachiger Befehle zu nutzen. »Es wird eingeschätzt, dass diese insgesamt nur etwa 100 englischen Wörter selbst von Nutzern ohne Englischausbildung ohne Mühe erlernt werden können« (Menhart/Stumpf 1987b, 17). Mit dieser Entscheidung erkannten das ZFT Robotron und die TU Dresden die Faktizität des Englischen als Lingua Franca der elektronischen Datenverarbeitung an. Ein Hintergrund dieser Veränderung dürfte sein, dass die qualifizierten Entwickler bereits seit Jahren mit aus dem Westen beschaffter Software arbeiteten, die häufig auf Englisch verfügbar war und sich so eine pragmatische Haltung im Umgang mit der ›westlichen‹ Sprache herausbilden konnte. Zudem war die Software prinzipiell für den NSW-Export konzipiert, schließlich verfügte die DDR hier tatsächlich auch über die Rechte und konnte im Prinzip DABA 1600 nach internationalem Recht lizenzieren.

Zwei Beispiele aus einem begleitenden Zeitschriftenartikel sollen die Lesbarkeit von SQL vergegenwärtigen. Eine Liste aller Schüler geordnet nach Klasse (aufsteigend) und Name (fallend) ließ sich mit der Eingabe von »SELECT NAME, SRN, KLASSE FROM SCHUEL ORDER BY KLASSE ASC, NAME DESC«<sup>323</sup> abrufen, und der Join-Verbund für eine Liste der Belegung der Klassenzimmer aufsteigend sortiert nach Schülern war mit der Befehlsfolge »SELECT NAME, SCHULE, KLASSE, ZIMMER FROM SCHULE, SCHUEL WHERE SCHULE, KLASSE = SCHUELKLASSE ORDER BY NAME ASC« abzufragen (Menhardt/Stumpf 1987a, 16). Diese Abfrage würde auch mit heutigen SQL-Datenbanken funktionieren, mehr als 30 Jahre später.

Die Sprach-Dokumentation reflektierte den damaligen internationalen Stand der Sprache SQL. So wurde darauf hingewiesen, dass Integritätsbedingungen für Einfügungen, also beispielsweise die Überprüfungen, ob Daten im richtigen Format eingegeben wurden, noch nicht implementiert seien. Externe Views, das heißt, fest auf dem Datenträger abgespeicherte Teilabfragen, waren 1984 ebenfalls noch nicht möglich. Der geringe Speicher des K 1600 erschwerte die Bildung von Ergebnismengen, die über mehr als zwei Tabellen hinweg reichen sollten. De facto musste eine Art Zwischenergebnis gebildet und gespeichert werden, mit dem weitergearbeitet werden konnte.

Weiterbildungsangebote bei Robotron-Projekt Dresden und im Robotron Anlagenbau Leipzig Schulungszentrum flankierten die umfangreiche Anwenderdokumentation ebenso wie einführende Produktinformationen in den wesentlichen Fachzeitschriften der DDR. Im Magazin *rechentechnik/datenverarbeitung* hieß es: »Der Weg zu den Datenbeständen, der bisher über die EDV-Abteilung, über das Schreiben von Auswertungsprogrammen, führte, wird mit der Datenbanksprache und der Dialogarbeit erheblich verkürzt. So können auch im Laufe der Arbeit entstehende Informationsbedürfnisse kurzfristig befriedigt werden. Schließlich wird die EDV-Abteilung auf diese Weise von Programmierarbeiten entlastet. Kritisch ist zu bemerken, daß noch nicht alle Anwender der Kleinrechentechnik die Vorteile einer derartigen Arbeitsweise erkennen. Der Dialogarbeit von Fachnutzern steht man noch abwartend gegenüber« (Menhart/Stumpf 1987a, 16). Der letzte Satz erinnert daran, dass für die unmittelbare Nutzung und Veränderung der Datenbasen durch Endnutzer der Widerstand der Systemadministratoren zu überwinden war, die um die ihnen anvertrauten Softwarefigurationen ›bangten‹. Zentral jedoch blieb das Versprechen: »Die Datenbestände sind für die Endnutzer in Tabellen abgelegt, so daß sie nur den Namen der Tabelle und der Spalte benötigen, um auf sie zuzugreifen« (ebd., 18).

---

<sup>323</sup> SRN ist die Abkürzung für ›Schülerregistrierungsnummer‹, es ist ein fortlaufender Schlüssel. ASC steht für Ascending und DESC für Descending.

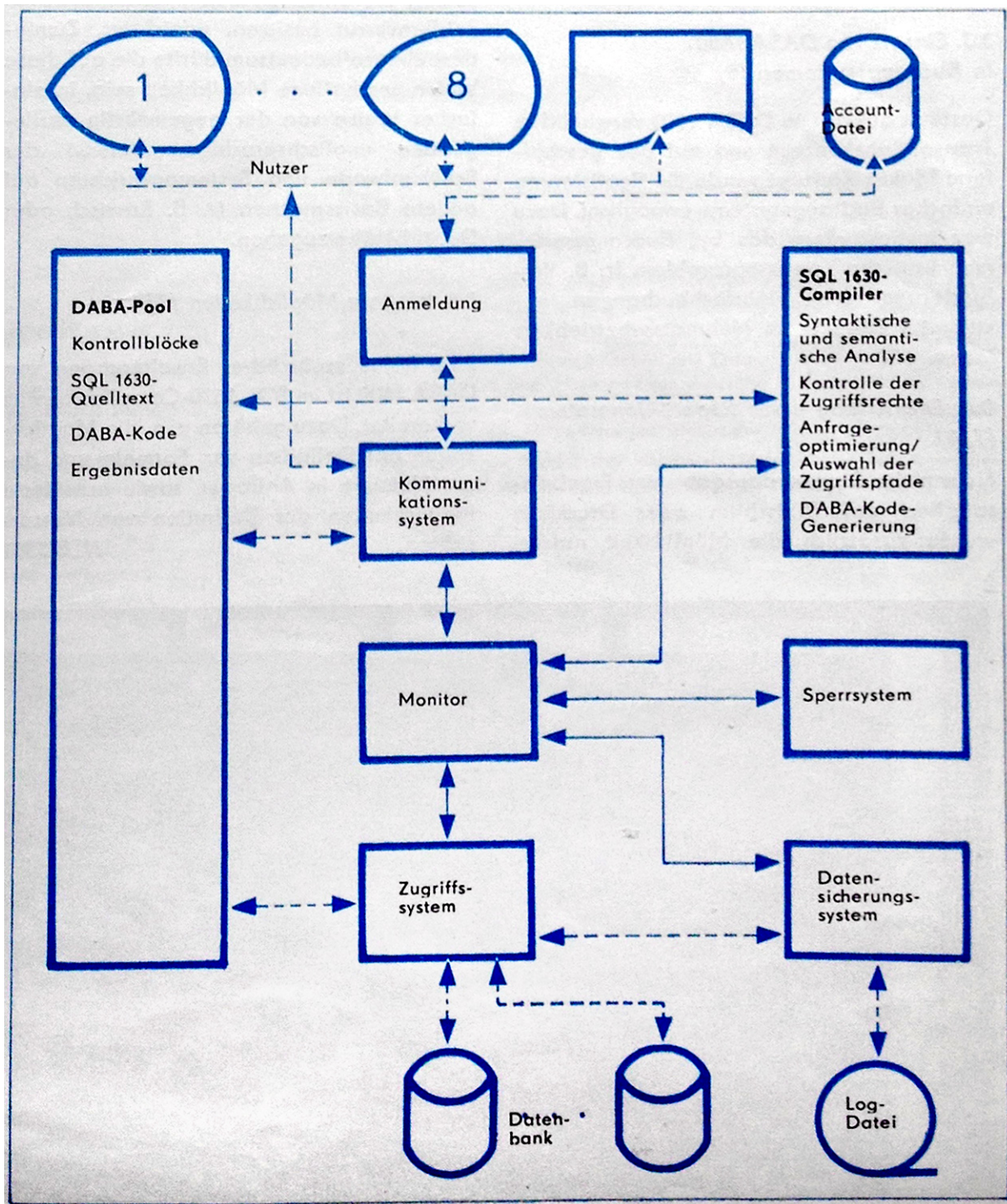


Abb. 81: Funktionsschema von DABA 1600 im Überblicksartikel in *Neue Technik im Büro* (Schubert 1985, 111).

Auch in *Neue Technik im Büro* erschienen Artikel, die DABA 1600 vorstellten, so zum Beispiel *Das relationale Datenbankbetriebssystem DABA 1600 – ein weiteres Ergebnis der Kooperation von Robotron und TU Dresden* von Prof. Dietrich Schubert (Abb. 81). Schubert wiederholte die Vision, welche als zentral für die Einführung der 16-bit Kleinrechner-technik in der DDR angesehen werden kann: die »Entdeckung« der Endnutzer, welche die Software unmittelbar einsetzen: »Solche Systeme, die heute im internationalen Maßstab eine rasch wachsende Verbreitung finden, zeichnen sich gegenüber den bisher in der Praxis eingesetzten Datei- und Datenbanksystemen dadurch aus, daß sie von Anwendern ohne

rechentechnische Spezialkenntnisse und ohne Unterstützung durch Rechnerspezialisten unmittelbar am Arbeitsplatz genutzt werden können« (Schubert 1985, 110).<sup>324</sup>

Wenn auch in der Dokumentation 1984 noch der Eindruck entstand, dass man sich damit auf dem aktuellen internationalen Stand (unter den Bedingungen der Kleinrechner-technik) bewegte, sahen die Autoren im Rückblick 2008 das Erreichte selbstkritischer: Oracle 2 lief auf der PDP-11, die dem K 1600 vergleichbar war. Doch bereits 1981 erschien Oracle 3, welches nach und nach auf verschiedensten Plattformen, d.h. Großrechnern, Kleinrechnern und Personalcomputern, unter verschiedenen Betriebssystemen lief (Demuth/Wiggert 2008, 282). In der DDR lief DABA 1600 ab 1986 zwar in der »Ausbaustufe 2.0« und verzeichnete circa 100 Installationen, blieb jedoch auf den K 1600 minicomputer als System beschränkt.

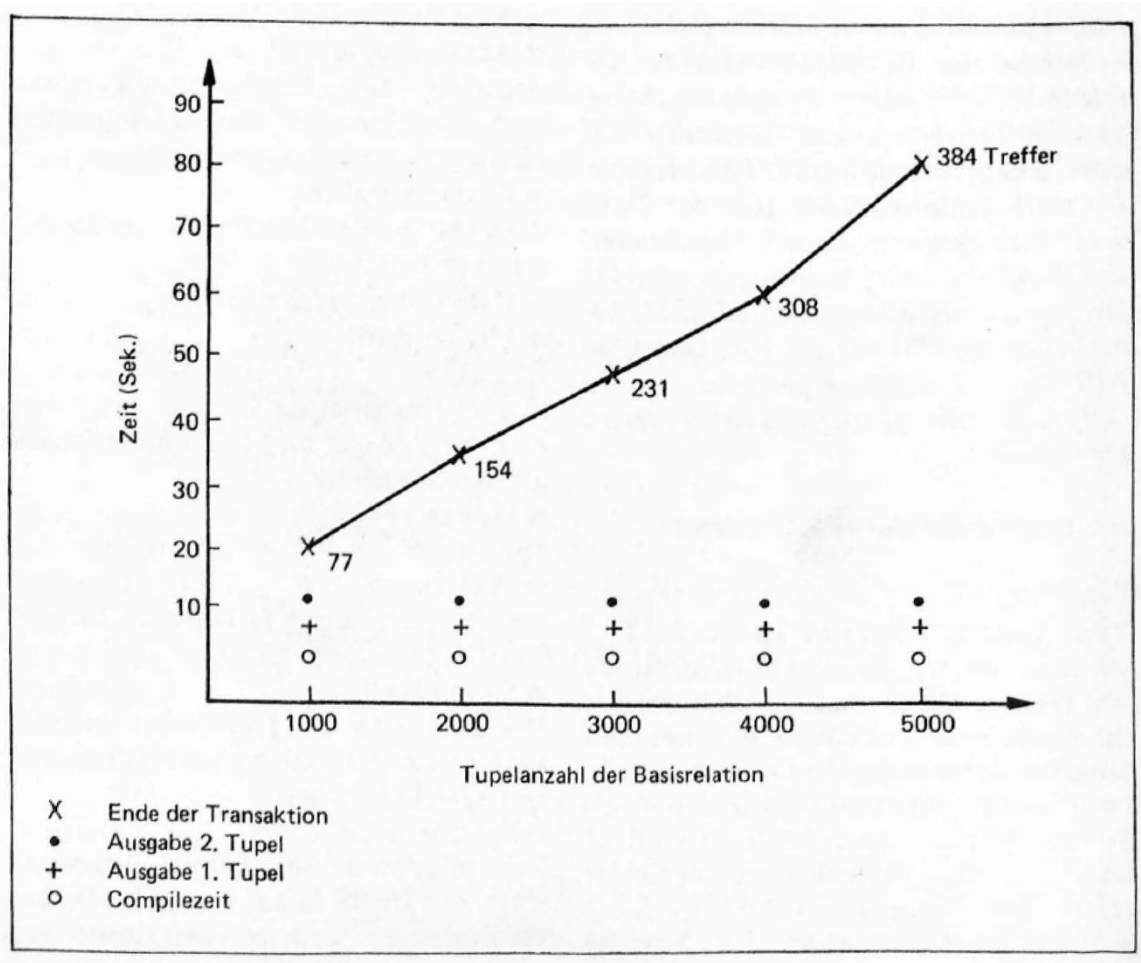


Abb. 82: Performance-Benchmark: Bei 2000 Tupeln (d.h. Zeilen) vergehen bis zum Ende der Transaktion im Schnitt 33 Sekunden und erzeugen 154 Treffer. (Menhart/Stumpf 1987c, 6)

Im Zuge der Produkteinführung kam es endlich zu den während des Forschungsprozesses an der TU Dresden noch nicht möglichen Evaluierungen der Optimierungskomponente im Zusammenspiel mit dem Datenbankbetriebssystem. »Erste gesicherte Ergebnisse von Zeitmessungen bestätigen, daß die Zugriffsmethoden und Techniken der Datenverwaltung für DABA 1600 richtig gewählt wurden. Dies

<sup>324</sup> Es handelte sich um eine Reihe von Artikeln in *rechentechnik/datenverarbeitung* (Menhart/Stumpf 1987b; Menhart/Stumpf 1987c; Menhart/Stumpf 1987d; Menhart/Stumpf 1987a) und in *Neue Technik im Büro* (Giebler 1980; Horn/Stumpf/Ziegenbalg 1981; Wiggert/Bulla/Ernst 1982; Heinemann 1985; Schubert 1985; Wiggert 1986).

drückt sich besonders in der wichtigen Funktion Wiederauffinden im Dialogbetrieb aus, wo die vorgegebene Zielstellung, Reaktionszeiten (Wartezeit zwischen zwei Mitteilungen des Systems) unter zehn Sekunden für durchschnittliche Abfragen zu erreichen, erfüllt wurde« (Menhart/Stumpf 1987c, 6), so die Selbsteinschätzung der Robotron Mitarbeiter (Abb. 82).

Weitere Forschungs- und Entwicklungsarbeiten bei Robotron wurden zugunsten des Datenbanksystems Robotron INTERBAS auf Eis gelegt, aber der Erfahrungsschatz konnte in das neue Projekt übergehen (Interview Bittner/Heinemann 2018, Fol 24). DABA 1600 lief jedoch aus Perspektive der zuständigen Leitung bei Robotron »nur« auf einem Kleinrechner, der dazu nicht IBM-kompatibel war. Die Firma hatte Größeres vor: INTERBAS. Tatsächlich müssen sich damals die Möglichkeiten, besonders die Geschwindigkeit von DABA 1600, gegenüber den Großrechnern und DBS/R als unterlegen dargestellt haben. Jedoch fanden die Kleinrechner des SKR eine zahlenmäßig deutlich größere Verbreitung als die Großrechner der ESER-Serie, sodass durchaus Potential bestanden hätte.

An der TU Dresden flossen die Erfahrungen an DABA 1600 in Demuths Vorlesung *Architektur und Implementierung von Datenbanksystemen* ein und damit unmittelbar an die Studierenden zurück. Ab 1988 arbeitete Demuth im Auftrag der TU Dresden an der Standardisierung von SQL in der DDR in einer TGL-Norm und nach 1990 auch im internationalen Standardisierungsgremium ANSI-ISO.

Mit der Verabschiedung der sozialistischen Planwirtschaft zugunsten kapitalistischen Markt-Geschehens endete mit der Produktlinie der Mikrorechner K 1600 auch die Ära des ersten eigenentwickelten relationalen Datenbanksystems der DDR. Dieses Ende schildern Birgit Demuth und Karl-Heinz Wiggert in einem retrospektiven Artikel mit einem knappen, trockenen Satz: »Als mit dem Ende des Kombinats Robotron auch keine K 1630-Rechner mehr produziert wurden, war der Einsatz von DABA 1600 gegenstandslos geworden« (Demuth/Wiggert 2008, 288). Dieses Gegenstandslos-Werden unterscheidet sich deutlich von den Kontinuitäten der Adaptionen Redabas, DBS/R und ALLDBS, welche in den Jahren nach 1989, insofern die Firmen und Institutionen fort bestanden, weiter im Einsatz waren. Die K 1630-Rechner jedoch schienen relativ schnell entsorgt worden zu sein. Für die DDR-Eigenentwicklung DABA 1600 gab es keine Nachfolgefirma. Die Nachfolger des Robotron ZFT konzentrierten sich auf die »westlichen« Datenbanksysteme Sybase und Oracle (Bittner/Heinemann 2018). Der Vorgang ist insofern bemerkenswert, als dass DABA 1600 mit dem SQL-Standard operierte, d.h. die resultierenden Datenbankdaten im Prinzip sehr gut auf andere SQL-Datenbankmanagementsysteme der damaligen Zeit portierbar waren. Vielleicht war diese Portabilität aber auch genau der Grund für das Verschwinden von DABA 1600.

Das Ende von Robotron muss nicht der einzige Grund für die Einstellung der Kleinrechner K 1600 gewesen sein – längst war bereits die Zeit der Personal Computer angebrochen, die ihre ganz eigenen Datenbank-Bedürfnisse und Datenbanksysteme mit sich brachten.

### 3.3.7 Fazit

Eine Darstellung des Wissenstransfers zwischen West und Ost könnte der Prämisse folgen, welche die »großen Ersten«, wie zum Beispiel IBMs System R, gegenüber nachfolgenden Entwicklungen hierarchisiert. Jedoch hat sich die Aufgabe der Medienhistoriographie gewandelt (Parikka 2012, 37ff; Haigh/Priestley 2015). Lagen frühere Ansätze darin, Medien überhaupt historisierbar zu etablieren, und gingen diese Ansätze damit einher, festzustellen, wer zuerst eine Erfindung gemacht hätte, so ist

statt der Erfindung inzwischen eher die Untersuchung von Emergenz und Einbettung technologischer Entwicklungen wesentliche Methode der Medientheorie geworden.

In diesem Kapitel wurden verschiedene diskursive Praktiken untersucht, die die Verschränkung von Forschung und Produktentwicklung bei DABA 1600 begleiten. Dabei wurde ein Perspektivwechsel vollzogen, der nicht nur die klassischen historischen Narrative über ›die DDR‹ verkompliziert, sondern auch die Pfadorientierung<sup>325</sup> der auf einzelne Erfinder fokussierten historischen Narration verlässt. Wenn auf Personen Bezug genommen wurde, dann vor allem mit dem Ziel, institutionelle Geflechte aufzuzeigen, in denen das Denken und Handeln bestimmten Bedingungen entsprach. Das Vorhaben unterlag einer Einschränkung: Anwendungsberichte, welche den medialen Gebrauch von DABA 1600 hätten näher beleuchten können, fehlen. So konnten hier aufgrund der Quellenlage in der Hauptsache die Motivationen der Forschenden, der Programmierer und der Institutionen nachgezeichnet werden, für eine Blick auf die Motivationen der Nutzer sei auf das Kapitel 3.2.3 *Dresden, Guben, Moskau, Rostock: Sachgebietsorientierte Programmiersysteme (SOPS)* verwiesen.

Erstens, ›Konzeptionelle Nachvollzüge relationaler Datenbanken‹: Die Ausarbeitungen zu DABA 1600 in der DDR ist als paradigmatisch zu lesen für die weltweit verstreuten konzeptuellen Nachvollzüge der relationalen Algebra Mitte der 1980er Jahre. Sie geben einen Einblick, welche Auswahlkriterien für technologische Entscheidungen eine Rolle spielten, wie Kooperation und Nicht-Kooperation den Forschungs- und Entwicklungsprozess begleiteten und welche personellen-institutionellen Überlagerungen und Verbindungen notwendig waren, um mit geringen personellen und finanziellen Ressourcen ein relationales Datenbanksystem von anfänglichen Forschungen zur Produktreife zu bringen. Paradigmatisch dürfte DABA 1600 deshalb für ähnliche Figurationen, wie zum Beispiel RISS (Relational Inquiry and Storage System) am Forest Hospital in Des Plaines, Illinois (USA) in Zusammenarbeit mit dem Massachusetts Institute of Technology im Jahr 1975,<sup>326</sup> und das DBMS DG/SQL auf einem Data General NOVA 1220-Computer von 1984<sup>327</sup> stehen, die heute nur noch dem Namen nach bekannt sind und zu denen wenig oder kein Archivmaterial mehr zugänglich ist. Paradigmatisch, weil es sich um vergleichsweise kleine Softwarefigurationen handelte, die nicht über Entwicklerteams in den Größenordnungen IBMs verfügten und die mit den zur damaligen Zeit relativ geringen Systemressourcen der Kleincomputer auskamen.

Zweitens, ›Wissenschaft als unmittelbare Produktivkraft‹: Spezifisch kennzeichnet das Projekt DABA 1600 in der DDR die äußerst enge Verflechtung von akademischer und industrieller Forschung. Sie ist auf wirtschaftspolitische Beschlüsse der SED zurück zu führen, welche ab Anfang der 1960er Jahre ›Wissenschaft als unmittelbare Produktivkraft‹ zum Aufbau des Realsozialismus objektivierten und im Rahmen der Planwirtschaft nutzbar machen sollten. Ihren Ausdruck fand diese ›unmittelbare Produktivkraft‹ in der ›Formalisierung materieller und geistiger Prozesse‹ mit dem Ziel der Automatisierung. Für die DDR versprach Automatisierung eine gesteigerte Intensität der Industrialisierung, um international konkurrenzfähige Produkte, zum Beispiel in Maschinenbau und Optik, und ab den 1980er Jahren verstärkt einheimische Konsumgüter produzieren zu können. Dieses Versprechen beinhaltete, dass die freiwerdende Arbeitskraft nicht in individualisierte Arbeitslosigkeit münden würde, sondern einen

---

<sup>325</sup> Zur Pfadorientierung von Geschichtsschreibung vergleiche Thomas Haigh's Kritik an der Darstellung von Ada Lovelace in der populären Technikgeschichtsschreibung Walter Isaacsons (Haigh/Priestley 2015).

<sup>326</sup> Siehe *RISS – A generalized minicomputer relational data base management system* (McLeod/Meldman 1975).

<sup>327</sup> DBMS DG/SQL war Eine Portierung des ›westlichen‹ INGRES auf Data General Minicomputer. Es war ein Nachfolge-Produkt des CODASYL-orientierten DBMS DG.

unterstellten gesamtgesellschaftlichen Nutzen hätte, der die Lebens- und Arbeitsbedingungen verbessern sollte. In dem ›Einerseits‹ der Exportfähigkeit und dem ›Andererseits‹ der Verbesserung der Lebensverhältnisse in der DDR enthüllt sich der Zielkonflikt, welcher die 1980er Jahre in der DDR prägte. In den Konfliktzonen versprachen elektronische Datenbanksysteme und die neue, relationale Logik einer der informationstechnischen Vermittler zu werden.

Die Prämisse von Wissenschaft als unmittelbare Produktivkraft führte zu zahlreichen Neuausrichtungen und Umstrukturierungen an der Technische Universität Dresden, während derer sich die neue Domäne der elektronischen Datenverarbeitung bzw. Informatik herauschälte. Diese Häutungen blieben nicht ohne Kosten für den akademischen Betrieb: eine zunehmend dringliche Auftragsforschung ging zu Lasten der Grundlagenforschung.

Drittens, ›Wissenstransfer‹: Die Softwareentwicklung an der TU Dresden kann als eine Laborsituation gekennzeichnet werden, die sich aus vergleichenden Studien und Metaanalysen speiste. Im kapitalistischen Westen entwickelte technologische Konzepte wurden in den Dissertationen ausführlicher Prüfung unterzogen, ggf. verworfen und erweitert. Unwillentlich personell und räumlich getrennt, entstand akademische Distanz. Die Geschichte ging mit ihren Subjekten ironisch um: So validierten die DDR-Wissenschaftler nicht allein ihre eigenen Forschungsansätze, sondern auch die der amerikanischen Kollegen. Dies soll nicht über die Nachteile für die DDR-Wissenschaftler hinwegtäuschen, denn der Besuch von akademischen Konferenzen und die Möglichkeit dort Kontakte zu knüpfen, blieben ihnen größtenteils verschlossen. Direkter Kontakt ermöglicht intensivere Wissensprozesse, im Vergleich zur distanzierten Rezeption anhand akademischer Journale, welche nicht vollständig, sondern nur zum Teil bezogen werden konnten. In der DDR konnte dieser Nachteil der Wissensproduktion partiell durch die räumliche Konzentration in Dresden ausgeglichen werden, die zu einem intensiven Austausch zwischen Forschung, Entwicklung und Anwendungspraxis führte.<sup>328</sup>

Wenn auch in ganz anderem Maßstab der Ressourcen und vermutlich mit unterschiedlicher Motivation, erinnert diese räumliche Konzentration an die Verquickung von Forschung, Lehre und Produktentwicklung im Silicon Valley und evoziert damit ein Spektrum aus Zentralisierung–Dezentralisierung, Planung–Emergenz und Ressourcen–Logistiken technologischer Entwicklungsprozesse. Die räumliche Konzentration war auch Ergebnis der bis Ende der 1970er auf Industrieproduktion konzentrierten Wirtschaftspolitik der DDR.

Viertens, ›Forschung und Entwicklung unter Embargobedingungen‹: Die nachgezeichnete Entwicklung von DABA 1600 kann aus dieser Perspektive als ein Beispiel dafür gelesen werden, ob und wie ein Wirtschaftsembargo greift. Zwei wesentliche Punkte fallen auf: Zum einen war auf dem Forschungsgebiet der Hardware für die DDR und den RGW das COCOM-Embargo deutlich wirksam, da Bauteile wie zum Beispiel elektronische Schaltkreise nicht bezogen werden konnten, sondern von Grund auf selbst entwickelt werden mussten. Selbst nachdem sie als Eigenentwicklung produziert werden konnten, waren die Stückzahlen zu klein, um rentabel zu arbeiten (Merkel 2019:19). Der zeitliche Abstand der Gerätegenerationen zeitigte zwar Auswirkungen auf die Softwareproduktion. Es relativiert sich dieser Abstand von drei bis sieben Jahren, wenn man andere historische Skalierungen aufruft: Aus der

---

<sup>328</sup> Hier konkurrieren zwei historiographische Leseweisen: Was einerseits als räumliche Konzentration gelesen werden kann, und damit Analogien zur Situation des Silicon Valley aufruft, ist aufgrund der Einbettung in eine Planwirtschaft häufig auch als zentralistische Top-Down Steuerung interpretiert worden, z.B. in (Sobeslavsky 1996, 82; Barkleit 2003, 137; Fritsche 2005, 118).



Perspektive eines Jahrhunderts war der technologische Abstand zwischen den entwickelten Ländern des Kapitalismus und dem DDR-Realsozialismus gering. Zum anderen stellte sich auf dem Gebiet der Software die Situation anders dar, da das Embargo den Wissenstransfer nur in geringem Maße behindern konnte: Hier waren Raubkopien, Adaptionen und vor allem Eigenentwicklung anhand öffentlich zugänglicher Publikationen des Westens auf dem aktuellen Technologiestand möglich, eingeschränkt durch die Investivkraft der DDR. Sprachliche Barrieren und fehlende internationale Vernetzung scheinen relativ problemlos kompensiert worden zu sein, solange der Zugang zu Fachpublikationen möglich war.

Dem abrupten, unvorhersehbaren Ende von DABA 1600, nach dem Abbruch der zugrundeliegenden Hardwareproduktion durch Robotron folgt ein unerwartetes Nachspiel. Den im Prinzip illegalen Adaptionen ›westlicher‹ Software als ALLDBS, ING\_DB und Redabas war eine stärkere infrastrukturelle Persistenz beschieden, als der Eigenentwicklung DABA 1600, da erstere mit vergleichsweise geringem Aufwand auf die nach 1990 fortgeführten West-Vorbilder (Oracle, Ingres, dBase etc.) übertragen werden konnten. Es bleibt zu vermuten, kann jedoch nicht belegt werden, dass aufgrund der SQL-Kompatibilität von DABA 1600 die Datenbestände relativ einfach auf ein anderes relationales Datenbankmanagementsystem übertragbar waren. Dies traf allerdings nicht in gleichem Maße für die Datenverarbeitungsprogramme zu, aus denen heraus die SQL-Abfragen gestellt wurden. Diese verschwanden mit der Verschrottung der zugrundeliegenden K 1600 Kleincomputer.

Fünftens, ›Dezentralisierung des Computings‹: DABA 1600 ist ein wichtiges Beispiel für die Geschichte der Softwareproduktion am Übergang vom Großrechner zum Kleincomputer. Der Historiker Paul Ceruzzi bezeichnet diesen Übergang im nordamerikanischen Kontext als »die abermalige Neuerfindung des Computers« (Ceruzzi 2003, 110).<sup>clxix</sup> Mit dem (immer noch raumfüllenden) Kleincomputer änderten sich die interne Struktur, die Programmierung, die Vermarktung und die Anwendungsgebiete. Die Datenverarbeitungsmaschine rückt näher an ihre Nutzer. Sie erhielten zunehmend direkten Zugriff auf die Struktur, die Modelle und die Inhalte der Datenbanken. Diese Tendenz zur Dezentralisierung ist auch für die DDR nachweisbar. Dominierten in den frühen 1960er Jahren noch Großrechner in Rechenzentren in den Bezirksstädten, verteilten sich neue Praktiken der Datenverarbeitung ab Ende des Jahrzehnt in den Betrieben und Organisationen des Landes. Diese Dezentralisierung der Datenverarbeitung erscheint heute als Folge sich ändernder Anwendungsbedürfnisse nach einem weniger vermittelten, unmittelbaren Zugang zu Daten, deren Speicherung und Verarbeitung. Folglich wurden vorhandene Kulturtechniken wie die Tabelle neu kontextualisiert oder wie im Falle der Transaktionen formalisiert und in Algorithmen, Modelle und Datenstrukturen überführt.

Sechstens, ›Datenbanken als Infrastruktur‹: In der emergenten Formung relationaler Datenbanksysteme wird aus infrastruktureller Perspektive deutlich, wie entstehende Technologie mittels Standards, Mitgliedschaften und Gewohnheiten vorkonfiguriert werden kann. Hierarchische und netzwerkorientierte Datenbanksysteme wie BASTEI und DBS/R waren Mitte der 1980er Jahre in der DDR bereits tief eingebetteter Teil der Softwareinfrastrukturen und ebneten den Weg für die relationale Logik der Datenverarbeitung. Deshalb konnte Birgit Demuth relationale Datenbanksysteme als Teil von Informationssystemen in ihrer Dissertation 1983 als »objektives Erfordernis« bezeichnen (Starke (Demuth) 1983, 144).

Für die Nutzung von DABA 1600 sahen die Entwickler großes Potential in der vergleichsweise einfachen Erlernbarkeit der grundlegenden Sprachelemente von SQL-1630, welches sich auf 100 englische Wörter beschränkte und für ungeübte Anwender innerhalb von zwei Stunden grundlegend erschließbar

sein sollte. So lässt sich, im Zuge von DABA 1600 in den Dissertationen, in den Anwendungsdokumentationen und den begleitenden Fachpublikationen eine konstante Anrufung der Endnutzer feststellen. Nicht mehr der Systemoperator als Servicemitarbeiter eines Rechenzentrums steht im Mittelpunkt der Überlegungen, sondern diejenigen, die direkt auf die Datensammlung zugreifen können sollen. Hier deutet sich eine Veränderung der Ordnungen des Wissens an, die noch an der Schwelle ihrer Verwirklichung stand.

Erst mit dem heutigen zeitlichen Abstand können wir konstatieren, dass die Hoffnung, die Abfragen würden durch Nutzer per SQL formuliert werden (siehe: Codd/Edgar F. 1970), sich nicht bewahrheitet hat. Mit der Verbreiterung der Nutzerschichten auf Alltagsnutzer, die keine Fachleute waren, sondern einfach über einen Personalcomputer oder Smartphone verfügen, war in den späten 1970er Jahren noch nicht zu rechnen. Heute stehen hinter (fast) jeder Website durch Programmschichten opaquisierte Datenbankabfragen, welche durch visuelle User-Interfaces von den Nutzern abgeschirmt sind. Nunmehr wenden sich die Figurationen nicht länger allein an die Systemadministratoren der Rechenzentren, sondern immer stärker auch an Endanwender und beginnen damit, eine der Zielsetzungen, die Edgar F. Codd für die relationale Algebra formulierte, zu realisieren.

Siebtens, ›Dezentrierung der Genealogien?‹: In den USA entwickelte Teil-Konzepte zur Realisierung der relationalen Algebra als Softwarefiguration wie Compiler-Optimierung, Drei-Schichten-Modell und Transaktionskomponenten zirkulierten international. Angesichts der konkreten geographisch verortbaren konzeptuellen Herkünfte fällt es dem Autor – nach dem Durchgang durch das Material – schwer, die angestrebte Dezentrierung der US-Geschichtsschreibung in dem Maße vorzunehmen, wie ursprünglich angestrebt, da tatsächlich die maßgebenden Konzepte relationaler Datenbankmanagementsysteme in den USA bei IBM entstanden. Der starke konzeptuelle Bezug auf amerikanische Forschungsergebnisse, zum Beispiel in Bezug auf Compilertechnik oder den Sprachumfang von SQL wurde in den Dissertationen stärker transparent, als in den anfangs gesichteten Veröffentlichungen der Fachzeitschriften oder in den Anwendungsdokumentationen.

Einher geht dies mit einer Übernahme ›westlicher‹ Argumentation in Hinsicht auf medienkulturelle Aspekte der (beabsichtigten) Nutzung, wie beispielsweise Wedekinds Auffassung, dass hierarchischen und netzwerkorientierte Ansätze zu kompliziert seien oder Codds Diktum, die Nutzer ins Zentrum der Überlegungen zu stellen. Diese mussten freilich durch DDR-eigene Erfahrung bestätigt worden sein, um wirksam werden zu können.

Achtens, ›Systemübergreifende mediale Praxeologien‹: Daraus ergibt sich eine neue Fragestellung, die eine medientheoretische und eine ökonomiehistorische ist: Wenn sich Kapitalismus und Sozialismus im 20. Jahrhundert konzeptuell unterschieden haben sollen, wieso glichen sich dann die eingesetzten Medien und Denkinstrumente der Datenverarbeitung und –speicherung in dem weitreichenden Maße, wie hier aufgezeigt? Es scheint, diese voreilige Einschätzung sei erlaubt, dass das mediale Konzept der relationalen Datenbank in der Lage war, Anwendungsbedürfnisse zu erfüllen, die über die Systemgrenzen hinaus reichten. Ein wesentlicher Unterschied der Mediennutzung lässt sich allein in den Rahmungen der Veröffentlichungen finden, der auch bereits im Abschnitt zu den SOPS sichtbar wurde: Während im Sozialismus volkswirtschaftlich mit ›Rationalisierung‹ argumentiert wurde, wurde im Kapitalismus die Datenbank in den Dienst betriebswirtschaftlich orientierter Konkurrenz genommen. Dies würde oberflächlich gesehen Auffassung stützen, die Technologie als politisch neutral einschätzen. Auf den zweiten Blick führt dies zu einer stärkeren Bedeutung der Kulturen ihrer Nutzung (und deren Geschichte), die in den Vordergrund tritt gegenüber dem Prozess der technologischen Genese.

## 4. Kooperative Medienpraktiken der Datenbank

Die vorliegende Studie hat vier wesentliche Desiderate bearbeitet:

1. Eine Präzisierung der Genealogien relationaler Datenbankenkonzepte und -praktiken.
2. Die Untersuchung von Datenbankpraktiken in der Industrieproduktion und Dienstleistung als Medien der Kooperation.
3. Historische Unterschiede und Gemeinsamkeiten medialer Figurationen der Kooperation in den unterschiedlichen sozio-ökonomischen Systemen Kapitalismus/Sozialismus bzw. West/Ost.
4. Ein für Datenbanken angemessenes Konzept von ›Information‹ und ›Daten‹.

Das Kapitel fasst die Ergebnisse der Materialsichtung zusammen, liefert eine Diskussion der Konsequenzen für die Geschichte des Computings und Software Studies, und erörtert zukünftiger Forschungsfragen. Wesentliche Punkte wurden durch Tabellen und Infografiken herausgestellt, welche die komplexen Wissenszusammenhänge in gedrängter Form darstellen sollen und zu diesem Zweck neu erstellt wurden.

### 4.1 Ergebnisse des untersuchten Materials

Die Studie wurde genutzt, um

1. einsetzend mit den 1960er Jahren die Herkünfte des relationalen Datenbankkonzeptes zu rekonstruieren,
2. die Praxis des Tabellierens ab dem 17. Jahrhundert als kulturtechnischen Vollzug von Information als Formation und als visuelle Metapher relationaler Algebra herauszuarbeiten,
3. einen weiten Bogen von Mitte des 15. Jahrhunderts in das 20. Jahrhundert schlagend, die koordinierenden Transaktionen dahingehend zu untersuchen, wie sie aus der betrieblichen Informationspraxis in Datenbankmanagementsysteme übertragen wurden,
4. spezifische Datenbankpraktiken des sozialistischen ›Ostens‹ und kapitalistischen ›Westens‹ anhand der modularen Softwarepakete SOPS für die DDR und SAP für die BRD in den 1960er bis 1980er Jahren historisch zu rekonstruieren, und
5. die Historie der Eigenentwicklung des relationalen Datenbankmanagementsystems DABA 1600 in der DDR der 1980er Jahre als Alternativhistorie zu den westlich dominierten Erzählungen aufzuzeichnen.

Zuerst, arbeitete das Kapitel 2.1 *Formatisierung – Strukturierte Daten in der relationalen Algebra* heraus, aus welchen Kontexten sich die Überlegungen zur relationalen Algebra speisten. Die historischen Darstellungen zur Genese des relationalen Modells konzentrierten sich bisher auf einen einzelnen IBM-Forscher. Eine Aufschlüsselung seiner Quellen zeichnete ein weitaus breiteren Kontext, der die überholte Erzählung eines einzelnen Erfinder-Genies modifiziert. So verdichtet sich eine Forschungsgenealogie, welche nicht allein auf IBM konzentriert ist, sondern sich als Forschungsnetzwerk innerhalb der USA erstreckt. Die diesbezüglichen Erkenntnisse des Kapitels sind in der folgenden Tabelle gedrängt zusammengefasst.

Forschungskontext	Institution	Personen
Information-Retrieval, Logikorientierte Abfragen, Relationalität	Cybernetic Research Data Project, US-Airforce/Rand Corp.	Roger E. Levien, Melvin E. Maron
Time-Sharing, Dialognutzung von Computern, Maschinenunabhängigkeit, Mengentheorie	Conversational Computing, ARPA/University of Michigan	David A. Childs, Frank Westervelt
Standardisierung allg. Anforderungen an Datenbanken	CODASYL, Database Task Group	Charles Bachmann und weitere
Betriebssystementwicklung (u.a. IBM OS/360)	Rand Corp., später IBM	George H. Mealy
Logistik, Systemarchitektur (IBM OS/370), Datenunabhängigkeit, später Transaktionen	US-Airforce, später IBM	Charles T. Davies
Relationale Algebra	IBM	Edgar F. Codd

Tabelle 4: Kontextuelles Netz der Emergenz relationaler Algebra. Dieses Netz war in der bisher verfügbaren Literatur nicht sichtbar und gibt neue Hinweise auf das Konzept ›Relation‹ (Autor).

Würde man dieses Wissenschaftsnetzwerk um die mathematischen und logischen Grundlagen erweitern, welche zu Anwendung kamen, so verknüpfte sich die nordamerikanische Geographie mit der Europas: Es wäre die Prädikatenlogik eines Gottlob Frege (Deutschland) zu nennen, ebenso wie die Arbeiten George Booles (England) zum algebraischen Logikkalkül, und zu der handlungsgeleiteten, pragmatischen Theorie der Bedeutung durch Willard van Orman Quine (USA).

Die Emergenz des relationalen Modells zeigt eine Phase der Abstrahierung und Akademisierung der Logiken der Datenarbeit an. Zwischen Maschinenlogik (physikalischer Aufbau symbolverarbeitender Maschinen) und humaner Logik (ausgedrückt durch tabellarische User-Interfaces) vermittelte eine neu eingezogene logische Ebene.

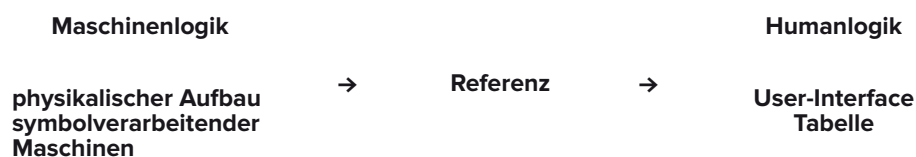


Abb. 83: Automatisierung koordinatorischer Praxen. Zentral die Referenz als neu eingezogene logische Ebene (Autor).

Diese logische Ebene automatisierte die bis dahin durch Menschen ausgeübte koordinatorische Praxis in einer spezifischen Form: Als relationale Logik erlaubte sie ein rechnendes Prozessieren mit Referenzen. Damit einher ging eine Verflächigung, eine Tableauisierung der Information. In den 1980er Jahren entstand so die neuen Möglichkeit, mit mathematisch begründbaren Regeln aus Relationen weitere Relationen logisch korrekt abzuleiten, und damit immer neue, logisch korrekte, Informationsmengen zu generieren. Damit schälten sich als zentrale Motive der automatisierten Informationsverarbeitung die Formatisierung von Informationen als ›flache‹, nicht-hierarchische Struktur, der Übergang vom Batch zum dialogorientierten Betrieb und die Automatisierung der Speicherverwaltung heraus. Dies gab

Anlass, abweichend vom Signal-Rauschen Informationsbegriff eine praxeologisch geprägte Auffassung von Information als Formation zu verfolgen.

Information als Formation zeichnet sich aus durch eine pragmatisch orientierte Referenzierung, indem für die Spalten Namen vergeben werden, welche im Verhältnis zu den Informationsobjekten in der Überschneidung von Spalte und Zeile, Referenz erzeugen. Die Praktiken des diagrammatischen Tabellierens gingen in die strukturierenden Datenmengen relationaler Algebra über. Das Kapitel verdeutlichte dramatisch, dass der Shannonsche Informationsbegriff mit seiner Fixierung auf Signal und Rauschen für Datenbankpraktiken zur Beschreibung heutiger Informationspraktiken mit Personalcomputern nicht genügt.<sup>329</sup>

Praxeologisch	Kybernetisch
Information als semantische Formation	asemantische Information
Namen	Signal-Rauschen
Referenz	Schwellenwert

Tabelle 5: Unterschiedliche Informationskonzepte (Autor).

Das folgende Kapitel 2.2 *Operationalisierung – Tabellieren* untersuchte Tabellen als kooperatives Denkwerkzeug. Durch die Kennzeichnung des Tabellierens als Kulturtechnik gelingt es, einen Zugang zu einer medienkulturtechnischen Sichtweise auf Datenbanken zu etablieren. Dieser Schritt war aufgrund der epistemologischen Verwandtschaft zwischen der relationalen Logik und der Tabelle angezeigt. In der medialen Praxis der Datenbankmanagementsysteme dient durchgängig die Tabelle als visuelle Metapher der Relation.

Tabellenpraxis	Wirkung	Wissensoperationen	Erstellungsweise
Wissen	N O R M A T I V	Synchronisierung und Normalisierung, Wissensübersichten, Speicher	Faktenorientierte Recherche und Kompilation
Statistik		Verknüpfung von Zeitreihen und Ontologien	Empirische Beobachtung
Transaktion		Verzeichnung und Synchronisierung von Handlungen im Verhältnis zu Basis-Informationsobjekten. Erzeugung von Analysedaten, Prozesskontrolle.	Zerlegung von Prozessen in Einzelabläufe
Mathematisch		Algorithmische Serien und deren Einzelwerte als Rechenhilfe	Industrialisierung von Denkarbeit im Modus der Arbeitsteilung

Tabelle 6: Epistemische Haupt-Funktionen der Tabellenpraktiken (Autor).

<sup>329</sup> Shannon selbst hatte acht Jahre nach Erscheinen seiner Informationstheorie im Editorial der IRE Transactions on Information Theory unter der Überschrift *The Bandwagon* davor gewarnt, seine Theorie der mathematischen Grundlagen entkleidet, anhand von Stichworten auf andere Wissenschaftsfelder zu verallgemeinern (Shannon 1956).

Mit verschiedenen historischen Einsätzen wurden vier Formen des Tabellierens erarbeitet, welche sich hinsichtlich Operationalität und Funktionalität unterscheiden, und jeweils verschieden gelagerte Referenzen auf die relationale Datenbank ergeben. Für alle Tabellenpraktiken konnte gezeigt werden, dass es einen übergreifenden Motivansatz zur Generierung neuen Wissens gibt: Die Beseitigung von Leere. Leere wiederum entsteht durch die Operation des Hinzufügens einer neuen Spalte. In allen Tabellen zeigte sich, wie die Leerstelle als Normativ auf das jeweils zu Speichernde wirkt, und zwar indem das Informationsmodell formatiert, wie Daten eingetragen werden können. Wir sehen hier, wie die Informationsverarbeitung als Normativeinschreibung fungiert, und so zum Gegenstand politischer Auseinandersetzung wird.

Der Übergang vom rigiden Raster der Tabelle zu den Ad-Hoc Abfragen relationaler Informationsmengen mittels Structured Query Language (SQL) macht uns zu Zeugen einer De-Stabilisierung der Anschauungsform der Episteme. Gemein ist beiden, der Tabelle und dem relationalen Konzept, das Informationsmodell, also der Entscheidung darüber, was Daten sind und was nicht.

Das Erkenntnisinteresse für den Abschnitt 2.3. *Koordination* – Transaktionen speiste sich aus einer Gegenwartsanalyse, dernach heute unter dem Stichpunkt ›Big Data‹ anonymisierte Massendaten gesammelt werden. Diese massiven Datensammlungen erlauben die Simulation von Prozessen, die Identifizierung von Gemeinsamkeiten in den Datenbeständen und individuelle Anrufungen der Nutzer, welche über Logins identifizierbar sind. Dabei kann, so die Prämisse der Big-Data Proponenten, die Kombination von Stammdaten und Transaktionsdaten durch Korrelation neues Wissen schaffen.<sup>330</sup> Durch die Automatisierung vormals arbeitsaufwendiger Transaktionsverzeichnung minimierten sich die Kosten einzelner Transaktionen. Es entstanden neue Transaktionsnetzwerke, deren Protokolle vom Lokalen ins Globale reichen und humane und nicht-humane Akteure miteinander koordinieren. Transaktionen erweisen sich somit als gekennzeichnet durch Nachvollziehbarkeit, Umkehrbarkeit, Abgeschlossenheit, Synchronizität. Sie basieren auf existierenden Praktiken der Information und dienen der Koordination humaner und nicht-humaner Akteure, wobei letztere jene semantischen Fehlstellen überbrücken, welche die Spannung aus Lokalem und Globalen in Softwarecode und Informationsmodell hinterlassen haben.

On-Line Analytical Processing, der Vorgänger heutiger Big-Data-Ansätze, stellte eine Aktualisierung und Erweiterung bereits vorhandener kaufmännischer Praxen der Datenauswertung, Simulation und Vorhersage, dar. Es versprach eine automatisierte Verknüpfung und Analyse großer Datenmengen verschiedener Herkunft. Somit kennzeichnet On-Line Analytical Processing als techno-informelles Konzept in den 1990er Jahren den Übergang zur Informationsgesellschaft (Castells 2010, 218–223). Bleibt zu klären, welches die medientheoretischen Folgen dieser transaktionalen Genealogien sind: Das computerisierte und automatisierte Transaktionskonzept ist durch eine reversible Historizität markiert. Es erlaubt Transaktionen unmittelbar rückabzuwickeln und sie vollständig zu wiederholen. Daraus folgt eine Lesweise von relationalen Datenbankmanagementsystemen und der darin gespeicherten Daten als beständig instabile Figuration, welche in möglichst engen Abständen automatisiert in den Zustand der Konsistenz gebracht wird.

---

<sup>330</sup> Ausgeführt sind diese Analysen der individuellen Anrufung und individuellen Datenkörper in *Epistemic Harvest – The electronic database as discourse and means of data production* (Hunger 2018). Für den Bereich der Werbung wurde durch Journalisten dargelegt, dass ein Nachweis zwischen individualisiertem Werbeaufwand und Kaufverhalten statistisch nicht erbracht werden kann (Frederik/Martijn 2019), es ist daher nicht auszuschließen, dass es sich bei dem gewonnenen Wissen zum Teil um Phantasma handelt.

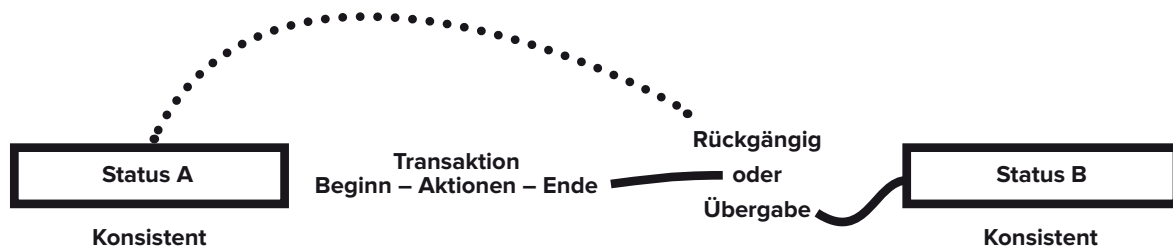


Abb. 84: Instabile Stabilität – reversible Historizität (Autor).

Eine weitere durchgreifende Folge von Transaktionsverzeichnung ist die massenhaft teilindividualisierte Ansprache von Individuen mittels Plattform-Medien. Diese zieht im Informationskapitalismus eine Ausweitung transaktionaler Wertschöpfung auf den zuvor unzugänglichen Bereich der sozialen Interaktion nach sich. Damit verbindet sich eine ständige transaktionale Präsenz der Appellation im (vormaligen) Privatraum. Die Unterscheidung in Privat und Öffentlich wird hinfällig. Im Sinne infrastruktureller Inversion bedeutet dies heute: Wo eine Transaktion ist, dort ist auch eine Datenbank. In der Mehrheit der Fälle ist dies eine relationale Datenbank.<sup>331</sup>

Das Kapitel 3. *Technoemergenz im Ost/West-Vergleich* untersuchte eine Reihe von Fallbeispielen. Hervorzuheben ist in der Einleitung der Abschnitt 3.1.4 *Software Adaptionen*, welcher eine neue Lesweise der Praxen des ›illegalen‹ Weitervertriebs von Software in der DDR vorschlägt. Diese werden stärker in einen Kontext des Programmierer-Alltags und der Fertigkeiten wie Reverse-Engineering, Debugging oder Re-Factoring gestellt. Die darauf folgende 3.1.5 *Übersicht über Datenbanksysteme in der DDR* gibt einen Überblick über die in der DDR verwendeten relevanten Datenbankmanagementsysteme und leitet die folgenden Fallbeispiele ein. Das Kapitel 3.2 *Robotron und SAP: Sozialistisches und kapitalistisches Enterprise Resource Management im Vergleich* blickt aus der Vergangenheit des Ostens in die Vergangenheit des Westens. Aus dieser sich mehrfach verlagernden Perspektivnahme nimmt das Kapitel die Entstehung von Produktionsmanagementsystemen und die Genealogien von Enterprise Resource Management, Customer Relationship Management, und Supply Chain Management ab den 1960er Jahren in den Blick.

Ausgehend von Stücklistenverwaltungen, und anknüpfend an die Transaktionstabelle, beobachtet das Kapitel die Umnutzung der spezifischen Datenmanagementsysteme Bills of Material Processor (BOMP) im Westen, beziehungsweise dessen Adaption namens Bankspeicherung Technischer Informationen (BASTEI) im Osten. Es erfolgte eine Medienumnutzung, die vom Anwendungsprogramm mit spezifischer Datenverwaltung zur generalisierenden Datenbank innerhalb der gleichen technischen Figuration führt. Die Medienumnutzung ermöglichte eine Geschwindigkeitsoptimierung der transaktionalen Informationsflüsse und damit eine Verdichtung der Produktion, und zwar in Bezug auf die Zusammensetzung der Produkte als auch die Zusammensetzung der Arbeit. Bestehende organisatorische Prozesse wurden innerhalb der technischen Figuration ›Datenbank‹ nachgebildet. Dabei wurden

<sup>331</sup> Neuere Technologien, wie Post-SQL, oder Blockchain können sich möglicherweise weiter ausbreiten, und relationalen DBMS Marktanteile abnehmen. Derzeit ist jedoch nicht abzuschätzen, inwiefern es sich um den Hype des Neuen, oder tatsächlich wirkmächtige Veränderungen handeln wird. Um ein Beispiel aus der Vergangenheit heranzuziehen: Dem Hype um objektorientierte Datenbanken in den 1990er Jahren folgte eine Teilimplementierung spezifischer objektorientierter Funktionen in relationale DBMS, und nicht die anfangs angenommene Verdrängung des einen durch das andere (vgl. Epstein 2013, 39).

Arbeitsabläufe tayloristisch zerlegt und in komplexe Softwareanwendungen codiert, innerhalb derer Daten eine zentrale Rolle spielten. Bestehende Lücken der Formalisierung wurden entweder durch sukzessive Anpassung der organisatorischen Prozesse überbrückt, oder durch die Nutzer informell korrigiert.

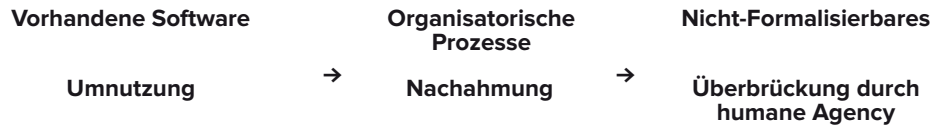


Abb. 85: Von der spezifischen Datenbankanwendung zur allgemeinen Datenbankssoftware (Autor)

Ein weiteres Ergebnis war die Darstellung der frühen Datenbankverwendung der Firma SAP. Diese stellt eine deutliche Präzisierung zur Frühgeschichte der derzeit wichtigsten Softwarefirma der Bundesrepublik dar. Damit wurde auch die Grundlage zu einem Vergleich mit den Entwicklungen in der DDR gelegt. Diese Art der Forschung in Ost und in West ist ein Beispiel dafür, wie blinde Flecken der Medienhistoriographie zu identifizieren und zu beheben sind.

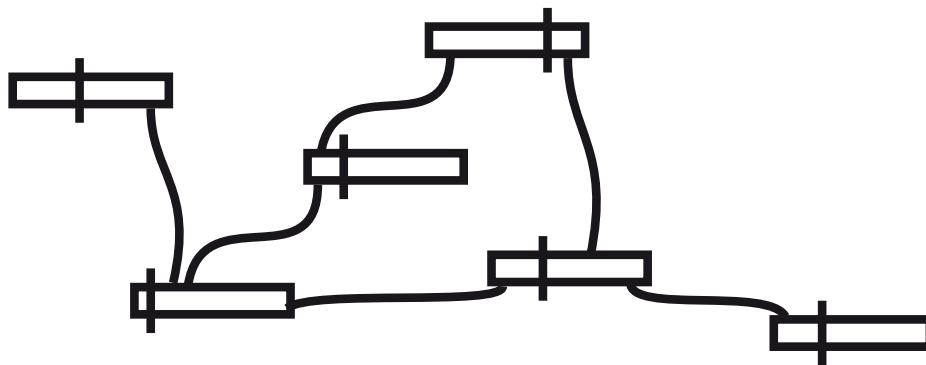


Abb. 86: Parametrisierbare Softwarefigurationen in SOPS und SAP. Die Schnittstellen von einem zum anderen Modul müssen standardisiert sein, um Informationen austauschen zu können. Jedes Modul kann über ein eigenes Informationsmodell verfügen, wobei die austauschenden Parameter der gleichen Art oder umwandelbar sein müssen (Autor).

Beide, SOPS im Osten und SAP im Westen, stellten innovative, wiederholbare, parametrisierte Figuren durch Softwaremodule (Finanzbuchhaltung, Personal, Materiallager, etc.) mit den dazugehörigen Praktiken der Handhabung dar. Sofern sie durch Parameter ausreichend an die lokalen Gegebenheiten angepasst werden konnte, und die beteiligten Nutzer bereit waren, die unweigerlich entstehenden Lücken zwischen Informationsmodell und Realität in Eigeninitiative zu überbrücken, fanden sie breite Anwendung. Technologisch waren die Unterschiede gering. Da hier Technologie immer auch als ein soziales Verhältnis gemeint ist, stellt sich die Frage nach den sozialen Gründen für die starke Ähnlichkeit der Softwarefigurationen.



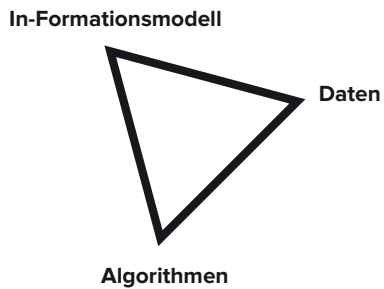


Abb. 87: Figurationen maschinellen Prozessierens sind verwoben mit der Trias aus Informationsmodell–Daten–Algorithmen, eingebettet in lokale und globale Infrastrukturen des Computings (Autor).

Das Informationsmodell, und die Entscheidung darüber, welche Aspekte von Realität als Daten aufgenommen werden, berühren politische Entscheidungen, da sie Fragen der Ein- und Ausschlüsse verhandeln. Es ist daher nicht, wie vielfach vereinfachend behauptet, ›der Algorithmus‹, welcher im Nachdenken über die gesellschaftlichen Folgen der Digitalisierung zentral gestellt werden sollte, sondern das Informationsmodell im Verhältnis zu Daten und Algorithmen. Diese Analyse ist umso eindringlicher, da gezeigt werden konnte, dass sie in verschiedenen politisch-ökonomischen Gebieten, in Ost und West zutrifft.

Jene Daten, die innerhalb eines Informationsmodells als computabel erachtet wurden, unterliegen einer doppelten Logik der Ein- und Ausschlüsse. Die Ausschlüsse führen im Zuge von Datenbanken, Big Data und von Künstlicher ›Intelligenz‹ einerseits zu Bias. Andererseits interessieren sich die Abfragen an Informationssoftwaresysteme nicht allein für das, was positiv als Daten erhoben wurde, sondern auch für Abweichungen vom Median innerhalb einer Proxy-Gruppe. Diese werden in den Behavioral Sciences als authentische Entäußerungen der Subjekte interpretiert (vgl. Chun 2018).

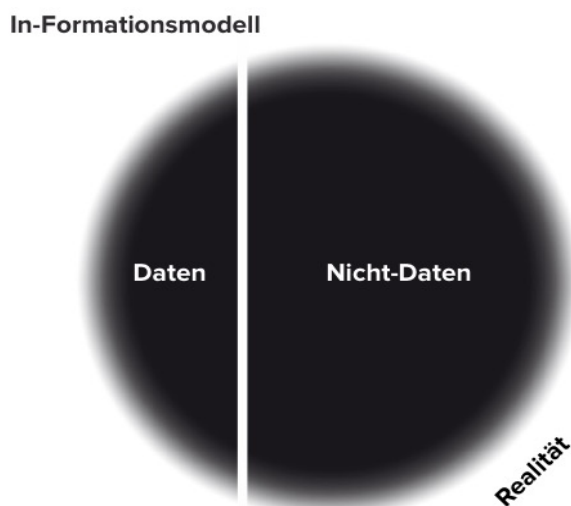


Abb. 88: Einschlüsse/Ausschlüsse von Realität im Informationsmodell. Daten werden durch Kategorisierung und Taxonomien im Informationsmodell definiert und der Überschuss bleibt »Nicht-Daten« (Autor).

Algorithmen können zwar spezifische Rechenvorschriften und Prozessabfolgen optimieren. Sie vermögen beispielsweise in Datenbanken die automatische Indexierung zugunsten schnellerer Suchmöglichkeiten zu verbessern. Sie versehen für die Entscheidungsnetze des Machine Learnings<sup>332</sup> die Berechnung mit jenen annehmbaren Geschwindigkeiten, welche den aktuellen Durchbruch künstlicher ›Intelligenz‹ prozessual ermöglichten. Doch sind es nicht Algorithmen, die die mediale Lage bestimmen. Anhand der Datenbankfigurationen sehen wir, dass es eine Trias aus Algorithmus–Daten–Informationsmodell ist, welche jene modularen Softwareanwendungen prägt, die das Funktionieren logistisch-informationeller Industriegesellschaften grundieren. Für Medientheorien der Software ist es daher fruchtbringend, sich von der Fixation auf ›den Algorithmus‹ als zentralen Ankerpunkt der Argumentation, zu lösen. Ähnlich zu hinterfragen ist die Zentralstellung von ›Netzwerken‹ in den Diskursen der 1990er Jahre, denn ohne Datenbanken an den Netzwerkknoten sind Netze dysfunktional.

Indem Daten aus den Praxen des Umgangs heraus gedacht werden, ergibt sich eine ›Materialität‹ der Daten. Stammdaten (Subjekt- oder Objektdaten) erweisen sich als eher statisch. Prozessuale Transaktionsdaten betreffen einzelne Vorgänge in großer Menge. Sie sind vergleichsweise dynamischer. Sie veralten schnell und unterliegen Entscheidungen über ihre Archivierung. Eine weitere ›Materialität‹ von Daten, die beobachtet werden konnte, bezieht sich auf Analysedaten. Diese zeichnen sich durch die Möglichkeit der Kombination topologischer und zeitlicher Ordnungen aus, welche neue Informationen generieren und beispielsweise durch On-Line Analytical Processing oder Big Data als epistemisches Material verwendet werden. Nicht-Daten schließlich sind der Schatten der Daten, sie konstituieren sich als Ausschluss aus dem Informationsmodell, als jener Anteil von Realität, der für die Computation nicht erfasst wird.

Wenn Daten aus einer infrastrukturellen Maintenance-Perspektive betrachtet werden, dann liegt der Schwerpunkt der Datenverarbeitung nicht mehr allein in ihrer Erhebung, sondern in der Erhebung, im Umgang, in ihrer Erhaltung und Konsolidierung. Deshalb gibt es Datenbanken.

Stammdaten	Transaktionsdaten	Analysedaten	Nicht–Daten
relativ statisch	dynamisch, veralten schnell	dynamisch, abhängig von Stamm- und Transaktionsdaten	
Subjekt, Objekt	Prozess	Auswertung und Simulation	Das aus dem Informationsmodell ausgeschlossene

Tabelle 7: Materialität von Daten in Datenbanken (Autor).

Das diskutierte Material kann in einem weiteren Punkt produktiv gemacht werden. In den logistisch-informationellen Industriegesellschaften ist eine Verlagerung, Dezentralisierung und logistische Neuzusammensetzung der Produktion zu beobachten. Diese wanderten in geographische Regionen, die nicht Teil der ursprünglichen industriellen Revolution des 19. und 20. Jahrhunderts waren, ein Prozess der als Globalisierung bezeichnet wird. Mediale Voraussetzung für Verlagerung, Dezentralisierung und neue Logistikern waren jene Softwarefigurationen, deren Entstehung durch modulare Betriebssoftware wie SAP und SOPS markiert sind. Vergleichend ist zu beobachten, dass im westlichen SAP R2, d.h. in

<sup>332</sup> Anthropomorphisierend als ›Neuronale Netze‹ in ›Künstlicher Intelligenz‹ bezeichnet. Eine kritische Einführung in die Terminologie leisten (Mackenzie 2017; Engemann/Jahn-Sudmann (Hrsg.) 2018; Pasquinelli 2019; Engster/Moore 2020).

der zweiten Version der Software, bereits eine skalierende Internationalisierung implementiert wurde. In SOPS und dessen Nachfolgeprojekten in der DDR ist diese nicht im gleichen Maße festzustellen. Sie bildeten keine internationalen Wirtschaftsstrukturen ab, wie es bei SAP der Fall war, sondern stellten allenfalls Übersetzungen in andere nationale Kontexte (z.B. der UdSSR) dar.

Mit dem Material des dritten Kapitels lässt sich die These bestärken, dass das Ende der DDR und des sozialistischen Ostens aus Perspektive von Datenbankfigurationen auch darin begründet lag, dass sie nicht darauf angelegt waren, im gleichem Maße wie der Westen auf eine global aufgefächerte Industrieproduktion umzustellen. Den westlichen Produzenten standen deutlich erweiterte Zugriffe auf Produktivitäts-, Arbeitskraft- und Preisdifferenzen zur Verfügung.

Im Zuge der Untersuchung war das durch die fordistische Fließbandproduktion geprägte Rationalisierungskonzept um ein transaktionsorientiertes Verständnis zu erweitern. Transaktionale Rationalisierung ist gekennzeichnet durch die Synchronisierung ursprünglich voneinander getrennter topologischer, geographischer und zeitlicher Domänen. Aus der Vereinheitlichung von Begriffen, Datentypen und Informationsstrukturen versprach man sich in der DDR sowohl eine stabilisierende Wirkung auf die innerbetriebliche Organisation, eine biopolitische Disziplinierung, als auch eine protokollarische Vereinheitlichung, welche sich günstig auf die zentral-orientierte Wirtschaftsplanung auswirken würde. Dass die erhofften Skalierungs-Effekte der Mehrfachverwendung verzögert, teilweise oder in anderer Form eintraten, war nicht DDR-spezifisch. Es ist ein verallgemeinerbares organisationstheoretisches Problem, welches im Zusammenspiel von Zentralisierung und Automatisierung der Informationsstrukturen im Zusammenspiel mit humanen Akteuren auftrat und auftritt. Wenn man den Übergang von einer am Ausstoß beziehungsweise am Fließband orientierten Rationalisierung zu einer transaktionalen Rationalisierung beschreibt, zeigt sich, dass es der DDR mit den SOPS zwar in einem gewissen Maße, und zwar auf Betriebsebene gelang, Elemente transaktionaler Rationalisierung einzuführen. Jedoch wurden deren skalierende Qualitäten im internationalen Maßstab in der DDR nicht ausgenutzt.



Abb. 89: Lokal verteilte Prozesse zusammengeführt im Datenbankmanagementsystem und durch Transaktionen synchronisiert. Rationalisierung durch Automatisierung der Informationsflüsse (Autor).

Das Kapitel 3.3 *Kalifornien, Technische Universität Dresden und Robotron Dresden: Das Problemorientierte Seminar »Datenbanken« und DABA 1600* rekonstruierte die Entwicklung und Programmierung eines relationalen Datenbankmanagementsystems anhand bereits eingeführter westlicher Konzepte und Überlegungen, die auf die DDR-Situation und DDR-Hardware angepasst und optimiert wurden. Erforscht wurde, auf welche Weise Entscheidungen fielen, wie institutionelle und personelle Engagements ineinander griffen, wie westliches Wissen im Osten akquiriert, aktualisiert und als eigene Forschungsleistung realisiert wurde. Die Eigenentwicklung des relationalen Datenbankmanagementsystems

DABA 1600 mit der Abfragesprache SQL an der TU-Dresden ist daher ein Beispiel für die weltweit vielen Nachvollzüge relationaler Konzepte.

Die in Dresden entwickelte relationale DABA 1600 markierte für die DDR den Übergang von den Großcomputern zu sogenannten Kleincomputern der K 1600/PDP-11 Klasse. Sie ermöglichte eine Dezentralisierung des Arbeitens mit Daten jenseits zentraler Rechenzentren. Als Nachspiel von DABA 1600 gilt es zu markieren, dass im Deutschland der 1990er Jahre, als der Osten zum Westen wurde, ihre Hauptentwicklerin Birgit Demuth die Abfragesprache SQL als Deutsche Industriennorm dokumentierte. Der Befund des vorhergehenden Kapitel bestätigt sich: auf technologischer Ebene waren die Ost- und Westsysteme durch vielfache Kontinuitäten geprägt. Trotz gewollter ideologischer Abgrenzung wurden im Zuge des Einheitlichen Systems Elektronischer Rechner (ESER) bzw. System der Kleinrechner (SKR) nicht nur technologische Konzepte, sondern auch organisatorische oder strukturelle Pragmatiken aus dem Westen übernommen, insofern sie sich für die DDR-Verhältnisse anpassen ließen.

Besondere als ›sozialistisch‹ zu markierende Anforderungen und Ausprägungen in der Entwicklung eines relationalen Datenbankmanagementsystems sind nicht zu verzeichnen. Damit sind die Trajektorien technologischer Konzepte durch die als Ost und West, beziehungsweise sozialistisch und kapitalistisch markierten Perioden hindurch, mit überraschender Deutlichkeit konsistent.

## 4.2 Konsequenzen für die Geschichte des Computings und die Software Studies

Den Ergebnissen folgend sollen im nächsten Abschnitt Konsequenzen, die aus der Recherche für die Geschichte des Computings und die Software Studies folgen, erörtert werden. Eine der Motivationen für diese Untersuchung war eine Dezentrierung der auf die USA fixierten Geschichte des Computings.<sup>333</sup> Sie liefert damit einen Beitrag zur vergleichsweise komplizierteren und kleinteiligeren Medien- geschichte Europas. Die Bedeutung der theoretischen Kybernetik der 1940er–1970er Jahre als Beschreibungsebene für datenorientierte Alltagspraxen ist kritisch neu zu bewerten. Stärker als bisher sind die in die Geschichte reichenden Praxeologien der Verwaltung und des Managements in den Vordergrund zu stellen, welche nach und nach computerisiert und damit automatisiert wurden. Diese These gehört zu den eindringlichen Ergebnissen der vorliegenden Studie und soll daher im folgenden Abschnitt eingehender diskutiert werden.

Dominanzverhältnisse im Wissenstransfer: Mit der hier vorgelegten Studie wird die stark auf Hardware konzentrierte DDR-Geschichte des Computings um eine Historie der Softwareproduktion erweitert. Der Bereich Software war von einer vergleichsweise größeren Durchlässigkeit von Forschungswissen geprägt als das Feld der Hardware. Die Schwierigkeit (aus Sicht des Ostens) lag im Erwerb von Quellcode, der sich auf den eigenen IBM-kompatiblen Hardwarearchitekturen kompilieren ließ. Gelang dies nicht durch Lizenzierung, griff man auf unlizenzierte, reverse-engineerte oder durch Industriespionage beschaffte Kopien zurück. Jenseits dieser Aneignungsverfahren unterhielt der Osten eine eigenständige Softwareproduktion, welche innovative Lösungen schuf. Diese waren im Bereich der Datenbankmanagementsysteme und vermutlich darüber hinaus durch einen starken Wissenstransfer aus dem Westen geprägt, denn der ›Eiserne Vorhang‹ ließ die akademische Rezeption des Westens im Osten zu. Mit

---

<sup>333</sup> Die Zentrierung auf die USA ist beispielhaft nachzuvollziehen in (Augarten 1984; Campbell-Kelly 2003; Ceruzzi 2003). Beiträge zu einer Dezentrierung lieferten unter anderen (Trogemann/Nitussov/Ernst (Hrsg.) 2001; Medina 2014; Peters 2017).

diesem doppelten Befund bestätigt sich die Dominanz des Westens und der USA im Bereich der Hardware- und Softwareentwicklung bis mindestens in die 1980er Jahre.<sup>334</sup> Eine Dezentrierung der Softwarehistoriographie entkommt diesem tatsächlichen Dominanzverhältnis nicht. Sie ist aber in der Lage wichtige Nuancen hinzuzufügen. Im Umkehrschluss bedeutet dies, dass global zahlreiches weiteres Material und lokal abweichende Praxen existierten, die als generelles Forschungsdesiderat verbleiben.

Kleines Land: An unerwarteter Stelle zeigen die Fallstudien zu den SOPS und DABA1600 einen eigenständigen Fokus – sie demonstrieren die Möglichkeiten und Limitationen der Softwareproduktion in einem ›kleinen Land‹ – der DDR. Deren wirtschaftliche Ressourcen waren beispielsweise vergleichbar zu Belgien, mit dem Unterschied, dass sie zu den sozialistischen, planwirtschaftlichen Staaten des RGW zählte.<sup>335</sup> Die DDR verfügte in den Jahren 1970 bis 1990 über mindestens sechs eigenentwickelte, fünf adaptierte und ein lizenziertes Datenbankmanagementsystem und kommt damit insgesamt auf mindestens 12 unterschiedliche Softwarefigurationen dieser Art. Diese wurden ab den 1970er Jahren sukzessive eingeführt, wobei nicht zwangsläufig eine Generation die nächste ablöste, sondern die verschiedenen Datenbankmanagementsysteme parallel zum Einsatz kamen. Im Umkehrschluss bedeutet dies, dass viele ›kleine Länder‹ über eine ähnliche Vielfalt von Datenbankmanagementsystemen verfügen dürften, abhängig von der Struktur der Hardware- und Softwareproduktion im jeweiligen Land. Wenn auch von einer historischen Dominanz der USA in Forschung und Entwicklung ausgegangen werden kann, ist diese nicht absolut, da zahlreiche einzelne Forschungsbeiträge international stattfanden.

Blinde Flecken der kybernetischen Perspektive: Von der Ost-West-Differenz ausgehend stand die Frage, welche Spezifika sich für den Umgang mit Datenbankensoftware in nicht-kapitalistischen Gesellschaften markieren lassen. Dafür gibt es Vorläufer. Die Beiträge Slava Gerovitchs *From Newspeak to Cyberspeak – a history of Soviet cybernetics* (2002), Eden Medinas *Cybernetic revolutionaries – technology and politics in Allende's Chile* (2014), und Ben Peters *How not to Network a Nation – The uneasy History of the Soviet Internet* (2017) zählen zu den motivierenden Momenten, sich des Themas anzunehmen.

Doch werden in Medinas Publikation Datenbanken oder die Arbeit mit Daten nicht erwähnt. Nur an einzelnen Stellen ist überhaupt von »Daten« die Rede, welche in »Regelkreisläufen« auftauchen (S. 203, 219), ohne konkreter darauf einzugehen. Eine Aussage wie Medinas, dass neue Technologie die Macht des Staates »rekonfigurieren« (S. 214) könne, erscheint angesichts der vorliegenden Untersuchung problematisch. So beobachtet Medina zwar Cybersyn als ›das Neue‹, übergeht in ihrer Beschreibung jedoch die nicht zu unterschätzende Persistenz bürokratischer Praktiken jenseits von Cybersyn. Diese dürften, folgt man den Erfahrungen der DDR, einen weitaus größeren Teil an der experimentellen Umsetzung von Cybersyn gehabt haben, als dies bisher erkenntlich wird.

Peters erwähnt eine geplante OGAS-Datenbank in einem Zitat Glushkows (S. 112f.) und fährt dann mit der Diskussion von Netzwerken fort, ohne weiter auf Fragen der Daten oder Datenbanken einzugehen.

---

<sup>334</sup> Abweichend davon: ab Anfang der 1980er Jahre spielte Japan mit den Herstellern Toshiba, Hitachi, Fujitsu, Mitsubishi und NEC eine prägnante Rolle für den Bereich der Integrierten Schaltkreise.

<sup>335</sup> Im Jahr 1989 betrug der GDP-Index der DDR 341 Punkte im Vergleich zu 491 für die BRD. Dazu im Vergleich andere ›kleine Länder‹: Belgien mit 353 Punkten und Niederlande mit 409 Punkten (Sleifer 2006, 50). Zur Debatte um die Aussagekraft dieser Zahlen siehe (ebd., 49–68). Hier sind sie vor allem aufgeführt, um die These zu unterstützen, dass ›kleine Länder‹ wie die DDR auf ähnliche Rahmenbedingungen in Bezug auf eine eigene Hardware- und Softwareproduktion trafen (vgl. auch Ball 2020).

Hierin zeigt sich ein blinder Fleck von Forschungsansätzen die von Theorien der Kybernetik ausgehen. Sie ergründungen die Persistenz bürokratischer Normative und deren Übergang in Softwareanwendungen nur unzureichend.

Im Unterschied dazu erzeugt der hier vorgelegte Beitrag eine andere Genealogie, als dies bei Gerovich, Medina und Peters und weiteren der Fall ist.<sup>336</sup> Für einen Ost-West bzw. ›Sozialismus‹/Kapitalismus-Vergleich ist es fruchtbarer, nicht auf die metatheoretischen Modelle, mit Kybernetik als Leitfigur, abzuheben sondern sich dezidiert mit den jeweiligen Praktiken auseinanderzusetzen. Daher widmete sich die Untersuchung intensiv den medienkulturtechnisch-technologisch ergründeten Materialitäten und Praxeologien der Informationsverarbeitung, welche aus dem Feld der Verwaltung, des Management und der Bürokratie stammen, und zieht daraus abweichende Schlüsse. Dies führt zu einem dreifachen Befund:

- 1.) Wenn es darum geht, die Geschichte des Computings im Osten zu ergründen und auf das spezifische Wechselspiel mit Fragen der sozialistischen Wirtschaft und der Planwirtschaft zu konfrontieren, reicht die kybernetische Sicht nicht aus, da sie zu stark Großkonzepte wie Cybersyn und OGAS beobachtet, die über ihre Planung kaum hinaus kamen. Für das Scheitern dieser Projekte werden staatliche, administrative und politische Widerstände ausgemacht, und jene Nuancen nicht erfasst, die sich in der alltäglichen Verwendung von Software wie den SOPS zeigen. Zukünftige genealogisch angelegte Forschungen sind daher stärker als bisher auf Fragestellungen jenseits der Operationsforschung, der Netzwerke und der künstlichen Intelligenz zu konzentrieren.
- 2.) Die vorliegende Studie zeigt eine ganze Reihe von Blindspots einer kybernetisch orientierten Historiographie. So war der Anteil der kybernetisch inspirierten Operational Research in der DDR im Verhältnis zu den transaktionsorientierten Sachgebietsorientierten Programmiersystemen deutlich geringer. Die Unterscheidung zwischen Sachgebietsorientierten Programmiersystemen und Verfahrenorientierten Programmierpaketen zeigt eindrücklich die Übermacht bürokratisch-kooperativer Genealogien und Praxen gegenüber den in kybernetischen Theorien basierten Verfahrenorientierten Programmierpaketen. Die Sachgebietsorientierten Programmiersysteme, welche in koordinierenden, transaktionalen und informationsorientierten Praktiken verortbar sind, waren von größerer praktischer Bedeutung in den Betrieben im Vergleich zu den Verfahrenorientierten Programmierpaketen welche die kybernetischen Seite von Kontrolle, Algorithmus, Optimierung, Netzwerk und Signalverarbeitung widerspiegeln. Zusammenfassend, war die datenbankbasierte Produktionssteuerung und Transaktionsverarbeitung in der DDR realitätsdurchdringender als die Funktion der (kybernetischen) Operational Research. Letztere stellte ein Versprechen der Optimierung dar, von dem sich schwer überprüfen lässt, ob es in Planung und Lenkung tatsächliche Wirkmacht entfaltetete. Hierzu wäre ein eigenes Forschungsprojekt, welches das Verhältnis aus Planungsinstanzen in der Planwirtschaft und Softwaregebrauch ergründet, notwendig.
- 3.) Der Fokus auf eine kybernetischer Metatheorie verdrängt jene Aspekte kooperativer Medien, welche konstituierend für Datenbanken als Medien sind. Natürlich involvieren auch relationale Datenbanken Aspekte von Kontrolle und Feedback. Doch sind diese kein Selbstzweck. Sie entstammen, wie aufgezeigt wurde, einer tayloristischen Genese der Arbeitsteilung. Während Kybernetik als Metatheorie nun innertheoretisch bleiben würde, erlaubt eine praxeologische Perspektive der Datenbank,

---

<sup>336</sup> Allerdings lieferte Peters ein äußerst informierte Kapitel zum schwierigen Verhältnis von Bürokratie und Planwirtschaft und zeigte darin unter anderem auf, dass es gerade die informellen Lösungswege waren, welche die formalisierten Planwirtschaftselemente stabilisierten. Damit leistet er einen historischen Beitrag zur Debatte um Formalisierbarkeit sozialer Prozesse und situier auch entsprechende kybernetische Vorstellungen als soziales Verhältnis (Peters 2017, 72–79).

außermediale Prozesse zur Kenntnis zu nehmen: Die Verdichtung von Arbeit mit Hilfe von Informationstechnologien kann über F. W. Taylor als die Arbeit betreffend, und damit als soziales Verhältnis gelesen werden.

Selbst wenn beide Prinzipien (kybernetisch und praxeologisch) ineinander übergehen, beispielsweise wenn Transportwege mit Hilfe von Operational Research Ansätzen optimiert werden, und die daraus folgenden Prozesse wiederum in Datenbanken abgelegt werden, oder wenn Informationen als Formation qua Feedback an ihren Ausgangspunkt zum Zwecke der Steuerung zurück geleitet werden, gilt dieser Befund.

Insofern ergänzt die vorliegende Studie das durch Medina und Peters gezeichnet Bild, denn sie beobachtet nicht nur die feedbackenden Informationskanäle zwischen staatlicher Leitung und betrieblicher Leitung, sondern die unmittelbar in Produktion und Dienstleistungen verwobenen transaktionalen Figurationen, welche ohne (relationale) Datenbanken nicht realisiert worden wären.

Die Problematik lässt sich noch weiter vertiefen. Welche Dimensionen kybernetisch orientierten Metatheorien<sup>337</sup> verborgen blieben, weil sie sich vorrangig zu algorithmischen und vernetzten Konzepten äußern und weitestgehend Fragen des Informationsmodells und der Datenspeicherung in Tabellen und Datenbanken ignorieren, soll im Folgenden in Form von Thesen zugespitzt werden.

- Die kybernetische Informationstheorie sieht vom Informationsgehalt ab und behandelt Informationen aus einer Perspektive der Signalübertragung. Damit erlaubte sie es, Informationsströme als prozessierbar und computerisierbar darzustellen.
- ›Information als Formation‹ hingegen ist als kulturtechnisch tief in die Menschheitsgeschichte reichende koordinierende Wissenspraxis markiert, welche Kooperation ermöglicht. Formatierende Informationsmodelle erlauben eine Zuschreibung von Referenz (Namen der Spalten) zu einzelnen Werten (in den Zellen der Tabelle), und deren bedeutungsabhängige Prozessierung im Computer im Zuge relationaler Algebra (=, >, <, AND, OR, NOT).
- Die epistemische Dimension der SQL-Abfrage `SELECT * FROM students WHERE age > 18` ist mit dem Shannon-Weaver-Signal-Noise-Informationsmodell unzureichend zu fassen.
- Nicht Algorithmen, sondern Informationsmodelle bestimmen die Lage – schlichtweg, weil *jede* aktuelle Softwarefiguration über Datenbanken verfügt, in denen das Informationsmodell als Erfassung computerisierbarer Realität konzeptualisiert ist.
- Das Internet und Techniken der Mustererkennung (sogenannte Künstliche Intelligenz) beziehen sich ideengeschichtlich auf kybernetische Ursprünge. Ihre isolierte Betrachtung ist jedoch nicht ausreichend, um das Feld des sozio-technischen Wandels zu beschreiben.
- Um heutige komplexe medialen Figurationen des Computings zu erfassen, sind die Herkünfte elektronischer Datenbanken aus den kulturtechnischen Operationalitäten der Tabelle und der Transaktionen neu zu positionieren. Softwareanwendungen wie SOPS und SAP entstanden nicht aus metatheoretischen Überlegungen der Kybernetik, sondern aus den Praxislogiken, welche die Ingenieure und Designer vorfanden und in Softwareanwendungen übertrugen.

Diese Repositionierung beinhaltet das Wissen darum, dass Tabellen, Statistiken und Bürokratie Teil gouvernementaler Praxen mit den einhergehenden biopolitischen Agenden waren. Indem auf die Anwendungspragmatiken in Betrieben und Büros fokussiert wird, eröffnet sich ein anderer Zugang auf

---

<sup>337</sup> Diese Aussage bezieht sich auf die Sekundärliteratur, von der hier einige der wichtigsten Monographien erwähnt sind: (Beniger 1986; Hayles 1999; Pias 2000; Pickering 2010; Medina 2014; Halpern 2014; Peters 2017; Müggenburg 2018; Hui 2019).

relationale Datenbankpraktiken, als dies vorrangig kybernetisch orientierte Science and Technology Studies vermögen würden.

### 4.3 Anschlussfragen

Der nun folgende letzte Abschnitt des Kapitels diskutiert eine Reihe von Anschlussfragen, die aus dem bisherigen Material und der Diskussion hervor gehen. Medientheoretisch orientiert sind im Folgenden die offenen Fragen zum Informationsmodell und zur Materialität von Daten. Eher soziologisch entwickeln sich zukünftige Forschungsfragen um Relationalität und Subjektform, sowie die Planungsdebatte und post-kapitalistische Automatisierung. Diese potentiellen Anschlüsse basieren auf der bisher vorgenommenen, materialorientierten Grundlagenforschung, und beleuchten, in welche Richtungen diese weiter entwickelt werden kann.

Erstens, ›Informationsmodell‹: Das Informationsmodell ist nicht allein bei relationalen Datenbanken anzutreffen, sondern auch in anderen Formationen des Computing, wie zum Beispiel der Künstlichen ›Intelligenz‹. An dieser Stelle soll nur auf Künstliche ›Intelligenz‹ eingegangen werden. Es zeichnet sich derzeit ab, dass Künstliche ›Intelligenz‹ nicht als einzelstehendes Experiment in die Welt der Berechnung eintritt, sondern eingebunden in Software-Anwendungen wie zur Fotobearbeitung (Berthouzot u. a. 2011), Spracherkennungs- und Übersetzungsprogramme (Brynjolfsson/Hui/Liu 2019), und Wirtschaftssoftware (Kohli 2018). Diese Einbettung bedeutet auch, dass zwischen den Künstliche ›Intelligenz‹-Modulen und den durch Informationsmodelle organisierten, Softwareanwendungen (mit inhärenter Datenbanklogik) ein Austausch durch Informationsübertragung stattfindet.

Selbst im Zuge der Klassifikation sind Künstliche ›Intelligenz‹-Modelle durch Information als Formation strukturiert. Den mathematischen Werten, welche die trainierten neuronalen Netze errechnen, werden Referenzen aus der Realität zugeordnet werden, sodass ein gewichtetes Netz eine Aussage erzeugen kann. Die zugrundeliegenden Trainingsdaten selbst werden klassifiziert, oft unter Zuhilfenahme schlecht bezahlter Clickworker. Dies soll kurz an einem Beispiel erläutert werden. Die *Tencent ML-Images* Datensammlung umfasst 17 Millionen Bilder in 11.000 Kategorien, welche auf der *ImageNet – A Large-Scale Hierarchical Image Database* (Li u. a. 2009; vgl. dazu kritisch: Crawford/Paglen 2019) und *Open Images* (Krasin u. a. 2017) basieren. Die Kategorien wurden durch Menschen, z.B. Clickworker, auf Bilder gelabelt, welche wiederum zur Konditionierung sogenannter ›Neuronaler Netze‹ verwendet werden können. Diese trainierten ›neuronalen Netze‹ können dann in Softwareanwendungen verwendet werden, um für beliebigen Bildern Ähnlichkeiten zu berechnen, und diese beliebigen Bilder automatisiert zu labeln, z.B. mit der ID 9051 »adulteress, fornicatress, hussy, jade, loose woman, slut, strumpet, trollop«. Neben der offensichtlichen Fraglichkeit der Kategorisierung an sich, ist zu sehen, wie hier Information in Formation gebracht ist und durch Referenzialität gekennzeichnet ist.

Mit Adrian MacKenzie kann zuspitzend darauf verwiesen werden, dass Vektoren eine mathematische Umwandlung des Tabellenrasters darstellen, nur dass diese nun in einen mehrdimensionalen latenten Raum übertragen werden. Wie stark das Prinzip der Räumlichkeit zur Wissensorganisation – wenn auch es sich der unmittelbaren menschlichen Anschauung entzieht – ist, zeigt MacKenzies Kommentar: »Sowohl in Bezug auf Infrastruktur als auch epistemische Kulturen abstrahiert und konkretisiert der Vektorraum Räume innerhalb von Daten« (Mackenzie 2017, 243).<sup>clxx</sup>



9049	n10275395	9046	loyalist, stalwart
9050	n09913593	9046	cheerleader
9051	n09772930	8560	adulteress, fornicatress, hussy, jade, loose woman, slut, strumpet, trollop
9052	n09633969	8560	wrongdoer, offender
9053	n10195593	9052	hypocrite, dissembler, dissimulator, phony, phoney, pretender
9054	n09756049	9052	abettor, abetter
9055	n10607478	9052	skinhead
9056	n09855433	8563	biographer
9057	n10017272	8563	disk jockey, disc jockey, dj

Abb. 90: Künstliche ›Intelligenz‹: Tabellarische Informationsstruktur für Tencent ML-Images mit den Spalten ›Kategorie‹, ›Kategorie ID‹, ›Übergeordneter Kategorie‹, ›Name‹, orientiert an ImageNet und WordNet (Tencent 2018).

Die Informationsmodelle Künstlicher ›Intelligenz‹ sind ebenso wie Information in Datenbanken durch Ein- und Ausschlüsse gekennzeichnet. Bias, also die Überbetonung bestimmter Klassen, oder das Fehlen anderer Klassen, ist dabei derzeit eines der offensichtlichsten Probleme (vgl. z.B. Noble 2018). Dies hängt auch mit der Art und Weise der Wissensproduktion zusammen, denn die Referenzen werden durch unterbezahlte Click-Worker, welche Bilder labeln, erstellt. Die damit zusammenhängenden Entscheidungen sind nicht nur kulturell abhängig, sondern zutiefst politisch, da sie Entscheidungen über die Computabilität von Realität sind (vgl. Mackenzie 2017; Steyerl u. a. 2018; Crawford/Paglen 2019; Pasquinelli 2019; Hunger 2019; Harvey/LaPlace 2019). Während die Geisteswissenschaften in den letzten Jahren aus feministischer und post-kolonialer Perspektive verstärkt auf diese Problematik hingewiesen haben (Noble 2018; Eubanks 2019; D’Ignazio/Klein 2020), ist dies bisher nur unzureichend in den Data Sciences wahrgenommen worden. Beispielhaft zeigte dies nicht nur die Existenz an sich von Datensets wie Tencent ML-Images, sondern auch eine heftige Debatte um den Artikel *Tracking historical changes in trustworthiness using machine learning analyses of facial cues in paintings* (Baumard u. a. 2020). Darin setzen die Autoren die Gesichtszüge aus einem Datensatz an Malerei seit 1500 in Korrelation mit ›Vertrauenswürdigkeit‹ – eine Verfahrensweise die als rassistisch, korrelationistisch und als schlicht unwissenschaftlich kritisiert wurde.<sup>338</sup>

Mit dieser Diagnose verbindet sich eine noch zu leistende Auseinandersetzung mit den kybernetischen Ursprüngen von Künstlicher ›Intelligenz‹ (vgl. Pinto 2015; Engster/Moore 2020) im Verhältnis zu ihren praxeologischen Zuordnungslogiken der Information als Formation (Schüttpelz 2016; Gießmann 2018). Ergeben sich hier möglicherweise neue Perspektivierungen, wenn die Ausgangsdokumente Künstlicher ›Intelligenz‹forschung nicht allein kybernetisch, sondern auch praxeologisch gelesen werden?

Eine daraus abzuleitende Forschungsfrage ist, wie bei weiterer Verschmelzung beider Technologien die Informationsmodelle der ›Künstlicher Intelligenz‹ mit den Informationsmodellen datenbankorientierte Softwareanwendungen interagieren, und welche Konstellationen von Macht sich in diesen Überschneidungen ausdrücken. Diese Fragestellung ist deshalb relevant, weil die Einschreibung von Dispositiven nicht, wie oft salopp behauptet, über ›Algorithmen‹ erfolgt, sondern durch ›Referenz‹ im Zuge des Informationsmodells. Es ist zentraler Ort von Technostruggles (Fiske 1996, 240).

<sup>338</sup> Siehe die Kommentare des Historikers Mateusz Fafinski <http://web.archive.org/web/20200928124945/https://twitter.com/Calthalas/status/1309403512867106816> und der Mathematikerin Sonia Balagopalan [http://web.archive.org/web/20200925135158/https://twitter.com/yet\\_so\\_far/status/1309475976376725504](http://web.archive.org/web/20200925135158/https://twitter.com/yet_so_far/status/1309475976376725504).

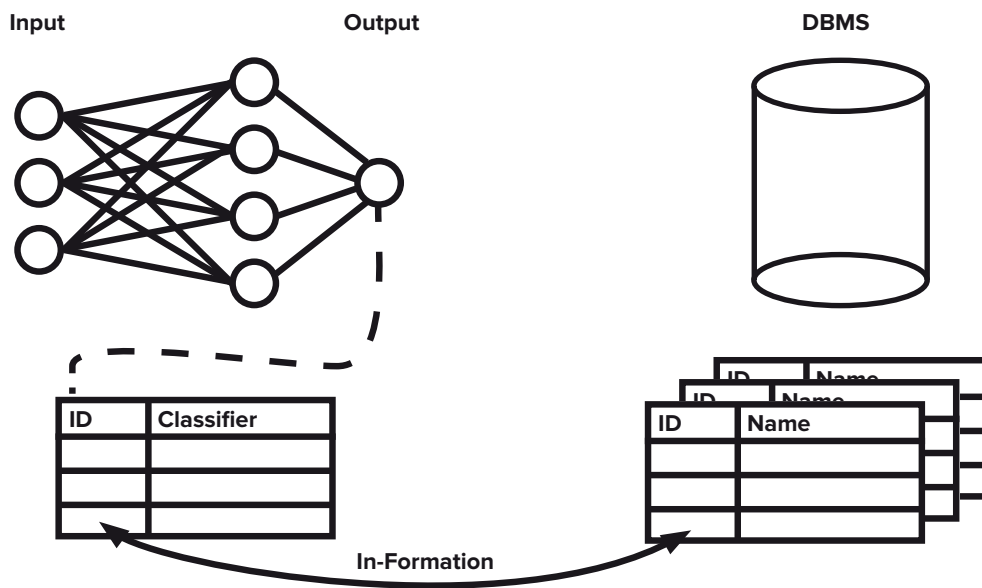


Abb. 91: Austausch von Information zwischen ›Neuronalem‹ Netzwerk (links) und Datenbankmanagementsystem (rechts). Während das ›Neuronale‹ Netz für kybernetische Operationalisierungen steht, ist bereits innerhalb der ›Künstliche Intelligenz‹ Softwarefiguration durch Zuordnung von IDs und Klassifikation Information als Formation am Werk (Autor).

Zweitens, ›Materialität von Daten‹: Die vorliegende Studie ist nicht allein als Untersuchung der Genealogien der relationalen Datenbanken zu lesen. Sie leistet auch einen Beitrag zur Materialität von Daten. Materialität meint dabei nicht das, was sich anfassen oder abtasten lässt, sondern die mit den Daten verbundenen mannigfaltigen Praktiken, die es erlauben, verschiedene Klassifikationen von ›Daten‹ zu unterscheiden. Ausgehend von einer engen Kopplung des Konzeptes ›Daten‹ an ›Information als Formation‹ konnten in der vorliegenden Studie verschiedene Materialitäten identifiziert werden:

- Stammdaten sind relativ statische Subjekt- und Objektbezogene Daten.
- Transaktionsdaten sind dynamischer und bilden Prozesse ab, sie werden ständig aktualisiert.
- Analysedaten führen zeitliche und topologische Ordnungen zusammen und fließen in neue Epistemologien ein.
- Nicht-Daten sind jene Anteile von Realität, die im Informationsmodell als unbedeutend oder nicht-computable abgespalten wurden.

Es gilt zu untersuchen, inwiefern anhand anderer Praktiken des Umgangs mit Daten, die hier nicht diskutiert wurden, zum Beispiel in Bibliotheken und Archiven oder in ›Soziale‹ Medien Plattformen, diese vorläufige Klassifikation zu ergänzen und modifizieren ist. Die informatorische Materialität von Daten würde eine medientheoretische Ergänzung zu anderen Konzepten davon, was Daten sind, leisten. So orientieren Couldry/Mejas ›Daten‹ soziologisch auf Information die erhoben und in Form gebracht wurde, »ausgerichtet auf Profiterzeugung« (Couldry/ Mejas 2019, XIII).<sup>clxxi</sup> Liu markiert medientheoretisch Daten als encodiert und strukturiert, als Information, und als gekennzeichnet durch die Trennung von Form und Inhalt, da die Form parametrisierbar ist und keinen Index auf den Inhalt mehr darstellt (Liu 2004). Gitelman ruft in Erinnerung, dass Rohdaten immer bereits aufbereitete Daten sind (Gitelman 2013). Dieser Umstand ist stärker in ein klassifizierendes Konzept von Daten einzubeziehen. Huis Konzeption von Daten als ›Gegebenes‹ ist gekennzeichnet durch deren Einbindung in »Digitale Milieux«, wobei er Daten, neben Metadaten, Datenformaten und Ontologien als eine von mehreren möglichen Formalisierungen »Digitaler Objekte« sieht (Hui 2016, XI). Auch bei Hui überwiegt eine

Sicht auf Daten, welche Struktur, beziehungsweise Information als Formation, hervorhebt. Wendy Chun macht darauf aufmerksam, dass in Datenaggregationen, die sich an Gruppenähnlichkeiten orientieren, ›Abweichungen‹ jenseits der Normalverteilung relevant sind. Dazu können sowohl ›Ausreiser‹ als auch Leerstellen zählen (Chun 2018, 76ff.). Auch Thylstrup weist auf den Aggregationscharakter von Daten, welche sie als »Spuren« kennzeichnet, hin. Sie markiert die Gefahren »toxischer« Datensets, die einem Prozess der Herstellung und Entsorgung unterliegen und schlägt vor, Datenmüll stärker in die Betrachtungen einzubeziehen (Thylstrup 2019).

Viele dieser Auffassungen von ›Daten‹ beziehen sich vorrangig auf die Anwendung von Daten auf Menschen und deren Subjektivierung in sozialen Prozessen der Kommunikation. Die vorliegende Studie hat im Unterschied dazu gezeigt, wie Daten in Bezug auf Prozesse (Transaktionen) und Objekte (Warenproduktion) erzeugt und genutzt werden. Es steht eine Überprüfung aus, ob beide Felder der gleichen ›Materialität‹ von Daten unterliegen, und welche Gemeinsamkeiten und Unterschiede es gibt.

Drittens, ›Planungsdebatte und post-kapitalistische Automatisierung‹: Ansätze wie Dietmar Dath *Maschinenwinter – Wissen, Technik, Sozialismus, eine Streitschrift* (Dath 2008), *The People's Republic of Walmart – How the World's Biggest Corporations are Laying the Foundation for Socialism* (Phillips/Rozworski 2019) und *Fully Automated Luxury Communism* (Bastani 2019) entwickeln die Idee, dass mit einer ›Vollautomatisierung der Welt‹ die Arbeitskraft befreit wird, und nicht länger für die Schaffung von Mehrwert verausgabt werden muss. Diese Überlegungen fanden ihr Pendant in der sogenannten Socialist Calculation Debate, welche Fragen der Preisbildung in einer Planwirtschaft in den 1920er und 1930er Jahren debattierte (vgl. Lavoie 1985, 2) und kürzlich durch Morozovs' Artikel *Digital Socialism? The Calculation Debate in the Age of Big Data* (2019b) neu aufgerollt wurde.

Die hier vorgelegte Studie der historischen sozialistischen und kapitalistischen Automatisierungen ruft für diese Debatten neue Fragestellungen auf:

- 1.) Ob das vielzitierte, jedoch kurzlebige Projekt Cybersyn ein geeignetes Fallbeispiel für Fragen der Automatisierung sozialistischer Preisbildung einerseits und Produktion andererseits ist, steht zu bezweifeln. Eine allein kybernetische Sicht, die sich in der bisherigen Forschung zu Cybersyn widerspiegelt, ist unzureichend, denn die praxeologischen Operationsketten sind langlebig und persistent. Die informatorischen Sachgebietsorientierten Programmiersysteme und kybernetischen Verfahrenorientierten Programmiersysteme der DDR waren für eine Zeitspanne zwei Jahrzehnten im Einsatz und geben daher einen besseren Eindruck sozialistischer Planung und Produktion.
- 2.) Die Verfahrenorientierten Programmiersysteme und andere automatisierte Planungsinstrumente der DDR wurden in der vorliegenden Studie nur gestreift. So gilt es, einen kompletten Diskurs um sozialistische Kybernetik und Operationsforschung der DDR medien-theoretisch und ökonomisch wiederzuentdecken. Eine ausführlichere Untersuchung könnte mit der Diskussion heutiger, ›kapitalistischer‹, softwarebasierter Planungstechniken kontrastiert werden.
- 3.) Die Arbeit der Staatlichen Planungskommission der DDR ist aus der Perspektive der Information als Formation und kybernetischen Steuerung und der Socialist Calculation Debate eine eigene medienwissenschaftliche Untersuchung wert, welche deutlich über die bisher vorliegenden historischen Veröffentlichungen von Medina (2014) und Peters (2017) hinaus geht.
- 4.) Jenseits der Frage der Preisbildung und der Socialist Calculation Debate: Die kooperativen Figurationen der Automatisierung sind in kapitalistischen und sozialistischen Industriegesellschaften untereinander erstaunlich austauschbar, und zwar weil sie auf tief in die

Industrieproduktion, Verwaltung und Bürokratie eingeschriebenen Operationsketten der Koordination basieren. Mit dieser Argumentation könnte das heutige SAP Softwarepaket auch für neue sozialistische Experimente eingesetzt werden. Dies eröffnet Potenziale für genossenschaftliches und nicht-kapitalistisches Wirtschaften, ähnlich dem Fediverse (Mansoux/Abbing 2020), welches auf verteilte, dezentrale und nicht-kommerzielle Alternativen zu gegenwärtigen ›Soziale‹ Medien Plattformen setzt.<sup>339</sup> Dies ist die optimistische Variante.

Weniger optimistisch gestaltet sich die Prognose, wenn, unter der Voraussetzung, dass man sie überhaupt in Kontinuität zum Staatssozialismus sehen möchte, die sozial-ökonomischen Entwicklungen in der VR China in und um Enterprise Resource Planning-Software beschrieben werden würden. Derartige Untersuchungen stehen derzeit aus (vgl. Ge/Voß 2009; Shi/Hoyt 2016).

Viertens, ›Relationalität und Subjektform‹: Die hier vorgelegten medientheoretischen Grundlagen sind ein Beitrag für aktuelle soziologische Debatten, die sich zu einseitig auf ›Soziale‹ Medien, also Medien der Kommunikation, konzentrieren. Hörl kennzeichnet die »Zerstörung des Bezugs, dessen Reduzierung auf die kalkulierbaren, kalkülisierbaren und verwertbaren Relationen, wie sie die Mathematik der Macht heute mit aller Konsequenz unternimmt.« als »neoliberal« (Hörl 2016, 40). Es ist weitergehend möglich, diese Tendenzen nicht allein als »neoliberal« zu lesen, sondern als neue, aktualisierte Stufe der im Kapitalismus üblichen extraktiven Wertverwertung (Postone 2003, 432–439; Couldry/Mejias 2019). Die Frage nach der Zerstörung des Sozialen durch dessen informatorische und automatisierte Integration in die Sphären des Verwertbaren bleibt hoch aktuell. Indem soziale Beziehungen in Form der Kommunikationsakte und Transaktionsakte aufgezeichnet und computabel gemacht werden, ist eine Extraktion zu konstatieren, welche in bisher unzugängliche Mikrokapellaren des Sozialen eindringt und neue Biopolitiken aufruft.

Einen Teil dieser Entwicklungen wurde in *Epistemic Harvest – The electronic database as discourse and means of data production* (Hunger 2018) diskutiert. Zur Subjektform wurde dort festgestellt, dass mit dem Hereinreichen von Kommunikation und Transaktionen mittels Datenbanken in die Privatsphäre sich der Status des Subjektes im Verhältnis zur Dualität Privat/Öffentlich ändert, da die Grenze zum Privaten beständig durchschritten wird. Die Subjektform ändert sich, denn es überlagert sich das Individuum mit dem Dividuellen (Raunig 2015; Ott 2015; Ott 2016). Diese Verschiebung erfolgt in einer spezifischen Form. Durch elektronische Datenbanken können Individuen dividuell repräsentiert werden, als Data Double (Lupton 2016) aufgespalten in ihre je nach Datenbank und Anliegen unterschiedlichen Teile. Das Unteilbare, das Individuum wird zum datenbankierten Teilbaren, dem Dividuum. Diese Verschiebung gilt es weiter zu verfolgen, und zwar ausdrücklich nicht nur für Medien der Kommunikation wie sie als Aufmerksamkeitsrente (Fisher/Fuchs 2015) und Kolonisierung von Daten (Couldry/Mejias 2019) theoretisiert wurden, sondern auch für Medien der Kooperation. Hier eröffnet sich als Alternative zum raunenden Surveillance-Diskurs (Zuboff 2019), ein Feld der Recherche und Theorie, welches nach dem sozialen Verhältnis von Informationsmodell, Daten und Algorithmen sucht.

Geht man der Frage ›Warum Datenbanken, warum jetzt?‹ nach, so entpuppt sich ein medientheoretisches Vorhaben einerseits als Grundlagenforschung und andererseits als politisches Unterfangen,

---

<sup>339</sup> Um Missverständnissen vorzubeugen: Eine Software allein kann den Prozess der kapitalistischen Wertverwertung (Heinrich 1999; Postone 2003) nicht außer Kraft setzen, dafür sind andere gesellschaftliche Veränderungen von Nöten.

welches es erlaubt, aktuelle Technostruggles einzuordnen. Die Frage, wie Deutungen von Realität in Technologie als soziales Verhältnis eingeschrieben werden, entzündet sich am Konzept der Information als Formation, welche das Gegebene, was Daten wird, einem normativen Prozess der Formatierung unterwirft.

#### 4.4 Coda: Deine Rücksendung wurde abgeholt.

Als Francis Hunger am Morgen aufstand, seine Kinder begrüßte und sich die Familie am Frühstückstisch traf, war er bereits mit zahlreichen Datenbanken in Berührung gekommen. Die Wasser-, Strom- und Heizungsähler zeichneten bereits auf, um ihre Daten per Funk an eine Datenbank zu übermitteln. Alle Hygieneartikel, die er verbrauchte, Müsli, Milch und Kaffee, die auf dem Frühstückstisch standen, waren über European Article Number (EAN) und Barcodes bereits mehrfach datenbankiert worden. Beim Hersteller wurde verzeichnet, was produziert und ausgeliefert worden war, und der Handel kalkulierte Preise, Ein- und Verkaufsmengen mithilfe der Datenbanken, die von den Kassierer\*innen an der Kasse beim Auslesen der Barcodes gefüllt worden waren.

Es klingelte. »Nicht jetzt!«, dachte er, denn seine Kinder mussten sich fertig für Schule und Kindergarten machen. Seine große Tochter fuhr eigenständig mit dem Fahrrad zur Schule, die Kleine brachte er in die Kita. Er fragte sich kurz, ob er an die Mittagessensbestellung beider Kinder gedacht hatte, die bei jeweils verschiedenen Anbietern per Internet in Datenbanken gespeichert wurden und eine lange, logistische Kette an Herstellung, Lieferung und Abrechnung nach sich zogen. Dann öffnete er die Tür. Der DHL-Bote brachte ein Paket für ihn und für zwei weitere Hausbewohner. Mit seinem Lesegerät scannte er die Pakete und aktualisierte den Stand der Transaktion, damit die Absender wussten, dass ihre Pakete übergeben worden waren. »Ich hätte hier noch eine Rücksendung«, sagte Hunger und übergab sie dem DHLer, welcher das Rücksendelabel einscannte, und mit einem tragbaren Mini-Drucker einen kleinen Belegzettel ausdrückte.

»Los, wir müssen zur S-Bahn.« ermunterte Hunger seine jüngere Tochter, die er zum Kindergarten begleitete, und überprüfte die Bahnverbindung schnell per App. Mit der App war er dauerhaft eingeloggt, über seine E-Mailadresse identifizierbar, sodass das Bahnunternehmen aufzeichnen konnte, welche Verbindung er gerade gesucht hatte, und ob er dafür online ein Ticket buchte. Machte er nicht. Denn er besaß ein Monatsabonnement, welches von den Kontrollleuren später per Lesegerät mit einer anderen Datenbank abgeglichen werden würde. Es gab eine leichte Verspätung und das war ihm ganz recht, denn so konnte er mit seiner Tochter entspannter losgehen.

Eine Reihe weiterer Datenbanken kreuzten seinen Weg, da er neben seiner künstlerischen Tätigkeit auch als Programmierer arbeitete und für seine Kunden neue Funktionen realisierte, oder Nicht-Funktionierendes reparierte. Zwischendurch hinterließ er auf Twitter ein Lebenszeichen aus 280 Buchstaben. Ein Zeichen, welches in Twitters' Datenbank abgelegt und abgerufen wurde, und welches die Aufzeichnung von Transaktions-Daten nach sich zog, wann immer jemanden die Nachricht affizierte sodass der/diejenige sie las, weiterleitete oder blockierte. Die von ihm und anderen programmierten Datenbanken wurden eingesetzt in der Warenproduktion, der Logistik, im Banking, für Marketingzwecke, zur Verwaltung von Kontakten, als Buchungs- und Archivdatenbank, zur Terminkoordination und im Zuge von Content Management Systemen für Websites.

Er war, wie viele hunderttausende Software-Programmierer\*innen, verantwortlich für die Erstellung des Informationsmodells, für die Ein- und Ausschlüsse, welche die Realität als Daten in der Datenbank verzeichneten oder als Nicht-Daten abspalteten. Er glaubte, das richtige zu tun, aber tat er es?

Hungers Tag soll nicht in Gänze geschildert werden. Die Datenbanken, davon circa 70% relationale, bestimmten sein Leben nicht, aber sie formten seine Handlungen, seine Entscheidungen, selbst die Zufälle mit, indem sie Normative automatisiert an ihn als Individuum zurück koppelten. Er ermöglichte sie und sie ermöglichten sein Leben. Manchmal bemerkte er es, meistens eher nicht.

Nachmittags überprüfte Hunger zum zehnten Male an diesem Tag seine E-Mails. Es gab nichts Neues, oder doch? Eine einzelne, neue Nachricht wurde geladen: »Deine Rücksendung wurde abgeholt« bestätigte ihm ein automatisierter, transaktionaler Prozess, welcher durch DHL datenbankiert worden war. Er schob die Nachricht in den Papierkorb, weshalb in der SQLite-Datenbank des Mailprogramms im die Nachricht betreffenden Papierkorb-Feld der Wert von 0 auf 1 gesetzt wurde.

Als er sich abends schlafen legte, korrespondierte auch dies mit einem Datenbankeintrag. Wenn die Rede davon war, dass seine Self-Tracking App die Schlafdaten »auf einem Server in der ›Cloud« speichere, so war eigentlich die Rede davon, dass die Daten in einer Datenbank verarbeitet wurden.

Datenbanken sind überall, man muss sie nur sehen.

# Bibliographie

## Interviews und Persönliche Kommunikation

- Anon. und Michael Zipf: *RE: Fwd: Fragen zu SAP*. Persönliche E-Mail (19.08.2019a).
- Anon. und Michael Zipf: *RE: Fwd: Fragen zu SAP*. Persönliche E-Mail (09.09.2019b).
- Bittner, Jürgen und Rolf Heinemann: »*Robotron hatte ja im Staat DDR die besondere Pflicht, die anderen zu versorgen*« *Interview mit Rolf Heinemann und Jürgen Bittner*. Interviewer: Francis Hunger, Dresden (14.11.2018).
- Chamberlin, Don: *Re: Contact Information*. Persönliche E-Mail (29.10.2016).
- Childs, David: *Re: STDS Questions (12-17)*. Persönliche E-Mail (08.10.2016a).
- Childs, David: *Re: Your work about the set theoretic data structure*. Persönliche E-Mail (21.03.2016b).
- Childs, David: *STDS Questions (12-17)*. Persönliche E-Mail (20.06.2016c).
- Codd, Sharon: *An Interview with Sharon Codd*. Interviewer: Francis Hunger, Boca-Raton (12.01.2016).
- Demuth, Birgit: »*Da war die DDR ein bisschen übermütig*« *Interview mit Birgit Demuth*. Interviewer: Francis Hunger, Dresden (28.11.2018).
- Merkel, Gerhard: »*Da wurden unsere Investitionen alle auf die Hälfte reduziert*« *Interview mit Gerhard Merkel*. Interviewer: Francis Hunger, Dippoldiswalde (16.12.2018).
- Salmond, Kent A.: *Oral History Interview about Charles Davies*. Interviewer: Francis Hunger, Los Gatos (14.06.2016).
- Uhlig, Matthias: »*In dieser Form haben die damals begonnenen Erfassungen die Zeiten überlebt*« *Interview mit Matthias Uhlig*. Interviewer: Francis Hunger, Leipzig (25.02.2019).
- Wloka, Uwe: *Interview mit Uwe Wloka*. Interviewer: Francis Hunger, Dresden (05.09.2019).

## Archivalien

- Bode: *[Entwurf Forschungsplan] Kombinat Robotron 1979/80 Abschrift für Genossen Peter Klick*. Manuskript, Dresden, Universitätsarchiv der TU Dresden, DFo 4658/I/01-19, (10.1979).
- Computerstunde Extra*. DDR-Fernsehen, Deutsches Rundfunk Archiv (20.10.1987).
- EboMath: *3 Übersichtslisten der bis zum 30 Juni 1951 bearbeiteten ingenieur-wissenschaftlichen Probleme*. Betriebsarchiv Carl Zeiss Jena, BACZ 11528, (21.09.1951).
- EboMath/Hass/Kul: *Tätigkeitsmerkmale für Entwicklungsbüro techn-mathematischer Probleme vom techn Hilfsrechner bis zum wissenschaftl Abteilungsleiter*. Betriebsarchiv Carl Zeiss Jena, BACZ 14966, (23.12.1953).
- Fernr. V.: *Entwicklung des Rechenbüros von Carl Zeiss*. Betriebsarchiv Carl Zeiss Jena, BACZ 16304, (1928).
- Kämmerer, Wilhelm: *Ausführlicher Abschlußbericht zur Forschungs- und Entwicklungsarbeit*. Betriebsarchiv Carl Zeiss Jena, BACZ 21563, (15.01.1956).
- Kerner, Immo: *Der Diplom-Mathematiker in der Industrie*. In: *Wissenschaftliche Zeitschrift der Karl-Marx-Universität Leipzig Betriebsarchiv Carl Zeiss Jena, BACZ 19247, (08.11.1959)*.
- Liebscher (Rektor), Fritz Wilhelm: *Leitlinien und Festlegungen für die Ausarbeitung der Fünfjahrplankonzeption Forschung in den Sektionen der TU Dresden für den Zeitraum 1981–1985*.

Manuskript, Dresden, Universitätsarchiv der TU Dresden, Akte Sektion Informationsverarbeitung TU Dresden: Fünfjahrpläne für die Forschung 1981–1985. Nr. 4163, (14.09.1978).

Minister für Elektrotechnik und Elektronik: *Ordnung der Planung und Leitung des VEB Kombinat Robotron*. BArch DG 10/116 (24.03.1969a).

Minister für Elektrotechnik und Elektronik: *Vorlage der Bestätigungsdokumentation zur Herbeiführung der Grundsatzentscheidung für das volkswirtschaftlich strukturbestimmende Investitionsvorhaben »Komplexe Erweiterung des VEB Kombinat Robotron*. BArch DG 10/116 (28.04.1969b).

Ministerrat der DDR: *Bericht über die Ergebnisse und die weiteren Maßnahmen bei der Gestaltung eines volkswirtschaftlich effektiven Produktions- und Exportprofils auf dem Gebiet der Rechen- und Bürotechnik im VEB Kombinat Robotron*. BArch DC 20-I/4/5241, 103. Sitzung des Präsidiums des Ministerrates der DDR (25.08.1983).

Ministerrat der DDR: *Beschluß betr Grundsatzentscheidung über das volkswirtschaftlich strukturbestimmende Investitionsvorhaben »Komplexe Erweiterung des VEB Kombinat ROBOTRON«*. BArch DC 20-I/4/2019 (23.07.1969).

Ministerrat der DDR: *Beschluß über die Grundrichtung für die Anwendung von elektronischen Datenverarbeitungsanlagen und Prozeßrechnern zur komplexen sozialistischen Automatisierung und Rationalisierung im Perspektivplanzeitraum 1971-1975*. BArch DC 20-I/3/788 (05.07.1970).

Ministerrat der DDR: *Beschluß zur weiteren Entwicklung und multivalenten Nutzung von Software in der DDR (einschließlich Materialien)*. BArch DC 20-I/3/2149 (04.04.1985).

Ministerrat der DDR: *Maßnahmen zur Beschleunigung der Entwicklungsarbeiten für die erforderliche Gerätetechnik und von Systemunterlagen sowie zur Erhöhung der Produktion von Geräten zur automatisierten Produktionsvorbereitung (CAD/CAM)*. BArch DC20/11199, auch: BArch DC20-I/4/5522 (20.12.1984).

Ministerrat der DDR und Politbüro des ZK der SED: *Grundlinie zur weiteren Beschleunigung der Entwicklung, Produktion und Anwendung der Mikroelektronik in der Volkswirtschaft der DDR*. BArch DC 20-I/3/1796, 15. Sitzung des Ministerrates der DDR (18.01.1982).

Ministerrat der DDR und Politbüro des ZK der SED: *Programm von Maßnahmen zur Entwicklung, Einführung und Durchsetzung der maschinellen Datenverarbeitung in der DDR in den Jahren 1964 bis 1970*. BArch DY 30/J IV 2/2/936 (23.06.1964).

SAP GmbH: *Neue Technologie der Basissysteme*. Kundeninformation, SAP History Communications (10.02.1985a).

SAP GmbH: *R/2 News International – Future Development Strategies*. Kundeninformation, SAP History Communications (07.06.1988).

SAP GmbH: *SAP Systeme R/2*. Broschüre, SAP History Communications (ca.1985b).

Schubert, Dietrich: *Planentwurf Forschung 1981–85 Profilierung auf Schwerpunktaufgaben und Leistungserhöhungen in Auswertung des X Parteitages der SED*. Manuskript, Dresden, Universitätsarchiv der TU Dresden, Akte Sektion Informationsverarbeitung TU Dresden: Fünfjahrpläne für die Forschung 1981–1985. Nr. 4163, (07.03.1981).

Sektion 08: *Staatliche Auflage 1983 für Aufgaben des Staatsplanes Wissenschaft und Technik (Berichtspflichtige Leistungen 1983)*. Manuskript, Dresden, Universitätsarchiv der TU Dresden, Akte Sektion Informationsverarbeitung TU Dresden: Fünfjahrpläne für die Forschung 1981–1985. Nr. 4163, (23.12.1982).

Sektion Informationsverarbeitung: *Forschungskonzeption 1981–85*. Manuskript, Dresden, Universitätsarchiv der TU Dresden, Akte Sektion Informationsverarbeitung TU Dresden: Fünfjahrpläne für die Forschung 1981–1985. Nr. 4163, (27.11.1978).

Sektion Informationsverarbeitung: *Komplexvereinbarung über wissenschaftlich-technische Zusammenarbeit zwischen dem VEB Kombinat Robotron und der Technischen Universität Dresden*. Manuskript, Dresden, Universitätsarchiv der TU Dresden, Akte Direktorium Forschung Dfo 4658/I/01-19, (12.06.1977).



- Sonnefeld, August: *Bericht an die Geschäftsführung*. Betriebsarchiv Carl Zeiss Jena, BACZ 16334, (1928a).
- Sonnefeld, August: *Die Arbeit unserer Rechner im Betriebe*. In: Zeiss Werkzeugzeitung Betriebsarchiv Carl Zeiss Jena (1921).
- Sonnefeld, August: *Tabelle zur Berechnung der Primärschablonen von Parabolspiegeln*. Carl-Zeiss-Jena, Betriebsarchiv, Z-369723, (13.09.1928b).
- Tschira, Klaus: *Der Übergang vom SAP-System R/2 zum System R/3*. Pressemitteilung, SAP History Communications (1991).
- Tschira, Klaus: *SAP in den 80er Jahren – Eine Betrachtung zum Dekadenwechsel von Klaus Tschira*. Internes Rundschreiben, SAP History Communications (12.1989).
- ZK der SED, Abteilung Forschung und Entwicklung: *Konzeption und Grundsätze der Organisation und Leitung für die Großforschung, Konzeption zur Einordnung von Wissenschaft und Technik die Leitung von Reproduktionsprozessen, Vorbereitung von Programmen für Großforschungsvorhaben, und Grundmodell eines Großforschungszentrums in Kombinat*. BArch DY 30/80181 (1968).

## Publikationen

- Abele, Johannes: *Modernisierung der Industriegesellschaft – Hochschulpolitik in der DDR*. In: Hänseroth, Thomas (Hrsg.): 175 Jahre TU Dresden – Wissenschaft und Technik. Studien zur Geschichte der TU Dresden Bd. 2, Köln, Böhlau 2003, S. 171–187.
- Adler, Paul S. und Bryan Borys: *Two Types of Bureaucracy – Enabling and Coercive*. In: Administrative Science Quarterly 41/1 (1996), S. 61.
- AG Rechentechnik: *Dateien*. In: Förderverein der Technischen Sammlungen Dresden. Website. <http://robotron.foerderverein-tds.de/download.html> [abgerufen am 14.03.2020].
- Agre, Philip E.: *Surveillance and Capture – Two Models of Privacy*. In: The Information Society 10/2 (1994), S. 101–127, DOI: 10.1080/01972243.1994.9960162.
- Ahrens, Wolfgang: *Anwendung des SOPS Bastei in der Chemiefaserindustrie*. In: Rechentechnik/Datenverarbeitung 13/1 (1976), S. 52–55.
- Alford, Leon Pratt und John Robert Bangs (Hrsg.): *Production Handbook*. The Ronald Press Company 1944.
- All Watched Over by Machines of Loving Grace Episode 2 - The Use and Abuse of Vegetational Concepts. (2011) <https://vimeo.com/groups/96331/videos/80799352>.
- Althusser, Louis: *Ideologie und ideologische Staatsapparate – Aufsätze zur marxistischen Theorie*, übers. v. Rolf Löper. Hamburg, VSA 1977.
- Aly, Götz und Karl Heinz Roth: *Die restlose Erfassung – Volkszählen, Identifizieren, Aussondern im Nationalsozialismus*. Berlin, Rotbuch Verlag 1984.
- Angé, Olivia und David Berliner (Hrsg.): *Anthropology and Nostalgia*. Oxford, NY, Berghahn 2015.
- Apitz, S., W. Saffian und A. Koslowski: *Erfahrung und Stand bei der Einführung eines integrierten Produktionsplanungssystems unter Verwendung der SOPS BASTEI und PLUS*. In: Rechentechnik/Datenverarbeitung 13/3 (1976), S. 28–33.
- Aristotle und Hans Günter Zekl: *Aristoteles' Physik – Vorlesung über Natur – Griechisch-Deutsch*. Hamburg, F. Meiner Verlag 1987.
- Arns, Inke: *Objects in the Mirror may be closer than they appear*. Dissertation, Humboldt-Universität zu Berlin, Philosophische Fakultät II, Berlin (22.11.2004), DOI: 10.18452/15154.
- Astrahan, M. M., J. W. Mehl, G. R. Putzolu, u. a.: *System R – Relational Approach to Database Management*. In: ACM Transactions on Database Systems 1/2 (1976), S. 97–137, DOI: 10.1145/320455.320457.

- Augarten, Stan: *Bit by Bit – An illustrated History of Computers*. New York, NY, Ticknor & Fields 1984.
- Augustine, Dolores L.: *Berufliches Selbstbild Arbeitshabitus und Mentalitätsstrukturen von Software-Experten in der DDR*. In: Zentrum Für Zeithistorische Forschung (Hrsg.): *Eliten im Sozialismus. Beiträge zur Sozialgeschichte der DDR*. Köln, Böhlau 1999, S. 405–433.
- Augustine, Dolores L.: *Red Prometheus – Engineering and Dictatorship in East Germany, 1945-1990*. Cambridge, MA, MIT Press 2007.
- Augustine, Dolores L.: *Werner Hartmann und der Aufbau der Mikroelektronikindustrie in der DDR*. In: *Dresdener Beiträge zur Geschichte der Technikwissenschaften* 28 (2003), S. 3–32.
- Autorenkollektiv, Rolf Gräßler und Hans-Dieter Freudenberg: *Problemorientierte Systemunterlagen im Betriebssystem OS/ES*. In: *Rechentechnik/Datenverarbeitung Beiheft* 12/1 (1975), S. 9–35.
- Autorenkollektiv und Walther Münch: *Einführung Themenheft ESER*. In: *Rechentechnik/Datenverarbeitung Beiheft* 11/2 (1974), S. 1–2.
- Autorenkollektiv: *Robotrontechnik - Die Geschichte der Computertechnik der DDR*. In: *Robotrontechnik*. Website. <http://www.robotrontechnik.de/> [abgerufen am 14.03.2020].
- Babbage, Charles: *On the Economy of Machinery and Manufactures*. New York, NY, Cambridge University Press 1832.
- Bachman, Charles W.: *Charles W Bachman Interview*. Interviewer: Thomas Haigh, Charles Babbage Institute/University of Minnesota, Minneapolis (26.09.2004).
- Balke, F., Joseph Vogl und Bernhard Siegert (Hrsg.): *Medien der Bürokratie*. Paderborn, Wilhelm Fink Verlag 2016.
- Ball, Joseph: *Does Socialism really lead to Economic Failure? The USSR and COMECON Eastern Europe before 1989*. In: *Journal of Labor and Society* 23/1 (2020), S. 87–110, DOI: 10.1111/wusa.12467.
- Barbari, Jacobo de: *Pacioli's Portrait*.
- Barkleit, Gerhard: *Mikroelektronik in der DDR – SED, Staatsapparat und Staatssicherheit im Wettstreit der Systeme*. Dresden, Hannah-Arendt-Institut 2000.
- Barkleit, Gerhard: *Mikroelektronik in Lehre und Forschung*. In: Hänseroth, Thomas (Hrsg.): *175 Jahre TU Dresden – Wissenschaft und Technik. Studien zur Geschichte der TU Dresden*. Bd. 2, Köln, Böhlau 2003, S. 259–269.
- Bastani, Aaron: *Fully automated Luxury Communism – A Manifesto*. London, New York, NY, Verso 2019.
- Batiz-Lazo, Bernardo und Douglas Wood: *An Historical Appraisal of Information Technology in Commercial Banking*. In: *Electronic Markets* 12/3 (2002), S. 192–205, DOI: 10.1080/101967802320245965.
- Batti et. al.: *RDBMS Plenary 1 – Early Years*. Interviewer: Burton Grad, Mountain View, CA, Computer History Museum, X4069.2007 (2007).
- Bauman, Zygmunt: *Liquid modernity*. Cambridge, UK and Malden, MA, Polity Press and Blackwell 2000.
- Bauman, Zygmunt und David Lyon: *Liquid Surveillance – A Conversation*. Cambridge, UK and Malden, MA, Polity Press and Blackwell 2013.
- Baumard, Nicolas, Coralie Chevallier, Julie Grèzes, u. a.: *Tracking historical changes in trustworthiness using machine learning analyses of facial cues in paintings*. In: *Nature Communications* 11/1 (2020), S. 4728, DOI: 10.1038/s41467-020-18566-7.
- Baxter, Rory: *How COPICS helped KTM grow in a harsh Environment*. In: *Production Engineer* 63/8 (1984), S. 32, DOI: 10.1049/tpe.1984.0169.
- Belke, Wolfgang: *Grundsätze zur Schaffung nationaler Standards auf dem Gebiet der Informationsverarbeitung*. In: *Rechentechnik/Datenverarbeitung* 22/1 (1985), S. 27–28.
- Bell, Daniel: *The coming of post-industrial Society – A Venture in social Forecasting*. London, Heinemann 1974.

- Beniger, James Ralph: *The Control Revolution – Technological and economic Origins of the Information Society*. Cambridge, MA, Harvard University Press 1986.
- Benthien, Claudia: *Haut – Literaturgeschichte, Körperbilder, Grenzdiskurse*. Reinbek, Rowohlt Taschenbuch Verlag 1999.
- Bergien, Rüdiger: *Im »Generalstab der Partei« – Organisationskultur und Herrschaftspraxis in der SED-Zentrale (1946-1989)*. Berlin, Ch. Links Verlag 2017.
- Bergin, Thomas J. und Thomas Haigh: *The Commercialization of Database Management Systems, 1969–1983*. In: *IEEE Annals of the History of Computing* 31/4 (2009), S. 26–41, DOI: 10.1109/MAHC.2009.107.
- Bertalanffy, Ludwig von: *Zu einer allgemeinen Systemlehre*. In: Bleicher, Knut (Hrsg.): *Organisation als System* Wiesbaden, Gabler Verlag 1945, S. 31–45.
- Berthouzoz, Floraine, Wilmot Li, Mira Dontcheva, u. a.: *A Framework for content-adaptive Photo Manipulation Macros – Application to Face, Landscape, and Global Manipulations*. In: *ACM Transactions on Graphics* 30/5 (2011), S. 1–14, DOI: 10.1145/2019627.2019639.
- Biener, Klaus: *Alwin Walther – Pionier der Praktischen Mathematik*. In: *RZ-Mitteilungen der HU Berlin* 18 (1999), S. 60–62, DOI: 10.18452/6275.
- Bittner, Jürgen: *DBS/R – Ein System der Praxis*. In: *Neue Technik im Büro* 28/2 (1984), S. 52–55.
- Bittner, Jürgen: *DBS/R Ausgabe 6 – Anwenderbedürfnisse bestimmen die Weiterentwicklung*. In: *Neue Technik im Büro* 30/2 (1986a), S. 174–175.
- Bittner, Jürgen: *DBS/R Symposien 1984, 1985 und 1987*. In: *Neue Technik im Büro* 30/6 (1986b), S. 173.
- Bittner, Jürgen: *Die Entwicklung von Datenbanksystemen für die Verwaltung strukturierter Daten im Kombinat Robotron*. In: Demuth, Birgit (Hrsg.): *Informatik in der DDR – Grundlagen und Anwendungen*. Bonn 2008, S. 268–279.
- Bittner, Jürgen: *Zur Aufgabenstellung des neuen Datenbankbetriebssystemes INTERBAS*. In: *Neue Technik im Büro* 31/6 (1987), S. 78–79.
- Bitton, Dina, David J. DeWitt und Carolyn Turbyfill: *Benchmarking Database Systems A Systematic Approach*. Very Large Data Bases San Francisco CA, Morgan Kaufmann Publishers Inc. 1983.
- Black, Edwin: *IBM und der Holocaust – Die Verstrickung des Weltkonzerns in die Verbrechen der Nazis*. Berlin, Propyläen-Verlag 2001.
- Bode, Bernd: *Lochkartentechnik*. Berlin, VEB Verlag Technik 1968.
- Bowker, Geoffrey C., Baker, Karen, Millerand, Florence, u. a.: *Toward Information Infrastructure Studies – Ways of knowing in a Networked Environment*. In: Hunsinger, Jeremy, Lisbeth Klastrup und Matthew Allen (Hrsg.): *International Handbook of Internet Research*. Dordrecht, Springer Netherlands 2010, S. 97–117.
- Bowker, Geoffrey C. und Susan Leigh Star: *Sorting Things Out – Classification and its Consequences*. Cambridge, MA, MIT Press 1999.
- Bowman, Larry F.: *SIGBDP 3 Performance measurement and data base design (Panel Discussion)*. ACM 75 – Proceedings of the 1975 annual conference ACM Press 1975, S. 49–50.
- Boyce, R. F., Don Chamberlin, M. M. Hammer, u. a.: *Specifying queries as relational expressions*. ACM Press 1973, S. 31–47, DOI: 10.1145/951762.951765.
- Braverman, Harry: *Labor and Monopoly Capital – The Degradation of Work in the Twentieth Century*. New York, NY, Monthly Review Press 1974.
- Bredenkamp, Horst, Birgit Schneider und Vera Duenkel (Hrsg.): *Das Technische Bild – Kompendium zu einer Stilgeschichte wissenschaftlicher Bilder*. Berlin, Akademie Verlag 2008.
- Bredow, Gabriel Gottfried: *Weltgeschichte in Tabellen nebst einer tabellarische Übersicht der Littäaturgeschichte*. Altona, Hammerich 1810.
- Brennan, Timothy: *Et sous la plage... ? Timothy Brennan im Interview mit Philipp Felsch zum Stand linker Theorie*. In: *Texte zur Kunst »Polaritäten«* (2016), S. 34–55.

- Brennen, Bonnie: *Qualitative research methods for media studies*. London, New York, NY, Routledge 2013.
- Bretschneider, Günther: *Die EDV als Instrument der Intensivierung des gesellschaftlichen Reproduktionsprozesses noch wirksamer nutzen*. In: *Rechentechnik/Datenverarbeitung* 11/10 (1974), S. 5–6.
- Bretschneider, Günther: *Zu einigen Aufgaben bei der Koordinierung des Einsatzes der elektronische Datenverarbeitung*. In: *Rechentechnik/Datenverarbeitung* 10/4 (1973), S. 5–7.
- Breuning, Lothar und Eckhardt Hoffmann: *Verträglichkeitsuntersuchung des SOPS BASTEI*. In: *Rechentechnik/Datenverarbeitung* 10/2 (1973), S. 29–32.
- Brie, Michael: *Lenin neu entdecken – Das hellblaue Bändchen zur Dialektik der Revolution & Metaphysik der Herrschaft*. Hamburg, VSA Verlag 2017.
- Bryer, R.A.: *Double-Entry Bookkeeping and the Birth of Capitalism – Accounting for the Commercial Revolution in Medieval Northern Italy*. In: *Critical Perspectives on Accounting* 4/2 (1993), S. 113–140, DOI: 10.1006/cpac.1993.1008.
- Brynjolfsson, Erik, Xiang Hui und Meng Liu: *Does Machine Translation Affect International Trade? Evidence from a Large Digital Platform*. In: *Management Science* 65/12 (2019), S. 5449–5460, DOI: 10.1287/mnsc.2019.3388.
- Bucher, Taina: *Objects of intense Feeling – The Case of the Twitter API*. In: *Computational Culture* (2013) Online Journal. <http://computationalculture.net/article/objects-of-intense-feeling-the-case-of-the-twitter-api> [abgerufen am 12.10.2019].
- Buchwald, Angela: *Die Ausbildung von Informatikern in Dresden – frühe Anfänge*. In: Demuth, Birgit (Hrsg.): *Informatik in der DDR – Grundlagen und Anwendungen*. Bonn 2008, S. 129–141.
- Buchwald, Angela: *Informatikausbildung in Dresden – 40 Jahre und mehr*. Manuskript, Universitätsarchiv der TU Dresden (22.04.2009) [https://tu-dresden.de/ua/ressourcen/dateien/integrierte-einrichtungen/ihd/40\\_jahre\\_inf?lang=de](https://tu-dresden.de/ua/ressourcen/dateien/integrierte-einrichtungen/ihd/40_jahre_inf?lang=de).
- Burkhardt, Marcus: *Digitale Datenbanken – Eine Medientheorie im Zeitalter von Big Data*. Bielefeld, Transcript 2015.
- Bush, Vannevar: *As We May Think*. In: *The Atlantic Monthly* (07.1945).
- Büttner, Gerhard, Reiner Dollner und Roger Kunitz: *Fertigungsbeleg 33*. In: *Rechentechnik/Datenverarbeitung* 9 (1981), S. 22–25.
- Campbell-Kelly, Martin, Mary Croarken, Raymond Flood, u. a. (Hrsg.): *The history of mathematical Tables – From Sumer to Spreadsheets*. Oxford, NY, Oxford University Press 2003.
- Campbell-Kelly, Martin: *From Airline Reservations to Sonic the Hedgehog – A History of the Software Industry*. Cambridge, MA, MIT Press 2003.
- Campe, Rüdiger: *Spiel der Wahrscheinlichkeit – Literatur und Berechnung zwischen Pascal und Kleist*. Göttingen, Wallstein Verlag 2002.
- Cantor, Georg: *Beiträge zur Begründung der transfiniten Mengenlehre*. In: *Mathematische Annalen* 46/4 (1895), S. 481–512, DOI: 10.1007/BF02124929.
- Cantor, Georg: *Grundlagen einer Allgemeinen Mannichfaltigkeitslehre – Ein mathematisch-philosophischer Versuch in der Lehre des Unendlichen*. Leipzig, Commissions-Verlag von B. Teubner 1883.
- Carroll, Lewis: *The Hunting of the Snark – an Agony in eight Fits*. New York, NY, Macmillan and Co. 1874.
- Castelle, Michael: *Relational and non-relational Models in the Entextualization of Bureaucracy*. In: *Computational Culture* (16.11.2013) Online Journal. <http://computationalculture.net/article/relational-and-non-relational-models-in-the-entextualization-of-bureaucracy> [abgerufen am 12.10.2019].
- Castells, Manuel: *The Rise of the Network Society*. 2. Aufl. Chichester, West Sussex, Malden, MA, Wiley-Blackwell 2010.
- Celko, Joe: *Joe Celko's Data & Databases – Concepts in Practice*. San Francisco, Morgan Kaufmann 1999.

Central Intelligence Agency: *Soviet RYAD Computer – A Program in Trouble (ER 77-10491)*.

Ceruzzi, Paul E: *A History of Modern Computing*. Cambridge, MA, MIT Press 2003.

Chamberlin, Don: An Interview with Donald D Chamberlin. Interviewer: Philip L. Frana, Charles Babbage Institute/University of Minnesota, Minneapolis (10.03.2001).

Chamberlin et. al., Don: *A History and Evaluation of System R*. In: *Communications of the ACM* 24/10 (1981), S. 632–646, DOI: 10.1016/0166-5316(81)90053-5.

Childs, David: *Description of a Set Theoretic Data Structure*. AFIPS '68 (part I) Proceedings December 9-11, 1968 New York, NY 1968, S. 557–564.

Childs, David: *Development of a Set-Theoretic Data Structure – Basis for Machine-Independent Information Management Systems*. ORA Project 07449, University of Michigan, Michigan (12.1970).

Childs, David: *XSP Technology*. In: XSP (04.07.2016d) Website. <http://xsp.xegeesis.org/> [abgerufen am 25.08.2016].

Chun, Wendy Hui Kyong: *Habitual New Media*. Lüneburg & YouTube (05.06.2014) <https://www.youtube.com/watch?v=p6TOICt8mtM> [abgerufen am 18.11.2019].

Chun, Wendy Hui Kyong: *Programmed Visions – Software and Memory*. Cambridge, MA, MIT Press 2011.

Chun, Wendy Hui Kyong: *Queerying Homophily*. In: Steyerl, Hito, Wendy Hui Kyong Chun, Florian Cramer, u. a. (Hrsg.): *Pattern Discrimination Lüneburg*, Meson Press 2018, S. 59–98.

Church, Alonzo: *Introduction to Mathematical Logic*, Bd. 1. Princeton University Press 1956.

Clark, Wallace: *The Gantt Chart*. New York, NY, The Ronald Press Company 1922.

Claus, Wolfgang: *Software-nachnutzung – Ziele und Mittel*. In: *Rechentechnik/Datenverarbeitung* 24/2 (1987), S. 5–8.

CODASYL Data Base Task Group: *A Survey of Generalized Data Base Management Systems*. New York, NY 1969a.

CODASYL Data Base Task Group: *Data Base Task Group Report to the CODASYL Programming Language Committee*. New York, NY, Association for Computing Machinery 1969b.

Codd, Edgar F.: *A Data Base Sublanguage founded on the Relational Calculus*. ACM Press 1971, S. 35–68, DOI: 10.1145/1734714.1734718.

Codd, Edgar F.: *A Relational Model of Data for Large Shared Data Banks*. In: *Communications of the ACM* 13/6 (1970), S. 377–387, DOI: 10.1145/362384.362685.

Codd, Edgar F. und Charles Bachman: *Discussion – Panel and Audience*. Proceedings of the 1974 ACM SIGFIDET (now SIGMOD) workshop on Data description, access and control: Data models: Data-structure-set versus relational. ACM Press 1975, S. 121–144.

Codd, Edgar F., Sharon Codd und Clynch T. Salley: *Beyond Decision Support*. In: *Computerworld* (26.07.1993), S. 87–89.

Codd, Edgar F.: *Data Base Management*. ACM Press 1975, S. 377, DOI: 10.1145/1499949.1500022.

Codd, Edgar F. und Chris J. Date: *Interactive Support for Non-Programmers – The Relational and Network Approaches*. ACM Press 1975, S. 11–41.

Codd, Edgar F.: *Derivability, Redundency, and Consistency of Relations stored in Large Data Banks*. IBM Research Report, RJ 599, IBM, San Jose/California (19.08.1969).

Codd, Edgar F., E.S. Lowry, E. McDonough, u. a.: *Multiprogramming*. Planning a Computer System – Project Stretch. London, New York, Toronto, McGraw-Hill 1962, S. 192–201.

Codd, Edgar F.: *The Relational Model for Database Management – Version 2*. Reading, MA, Addison-Wesley 1990.

Cohen, Malcom S.: *On the Feasibility of a Labor Market Information System – Final Report for Period July 1, 1970-June 30*, Bd. 3. Ann Arbor, University of Michigan/Institute of Labor and Industrial relations 1974.

- Colbert, Jean-Baptiste: *Lettres, Instructions et Mémoires de Colbert*, Bd. II/2, hrsg. v. Pierre Clément. Paris, Imprimerie Imperiale 1865.
- Conrad, Lisa: *Organisation im soziotechnischen Gemenge – mediale Umschichtungen durch die Einführung von SAP*. Bielefeld, Transcript 2017.
- Copeland, D.G., R.O. Mason und J.L. McKenney: *Sabre: the development of information-based competence and execution of information-based competition*. In: *IEEE Annals of the History of Computing* 17/3 (1995), S. 30–57, DOI: 10.1109/85.397059.
- Copeland, Duncan G. und James L. McKenney: *Airline Reservations Systems – Lessons From History*. In: *MIS Quarterly* (1988), S. 353–370.
- Cosentino, Christine, Wolfgang Ertl und Gerd Labrousse (Hrsg.): *DDR-Lyrik im Kontext*, Bd. 26. Amsterdam, Rodopi 1988.
- Couldry, Nick und Ulises Ali Mejias: *The Costs of Connection - How Data is colonizing Human Life and appropriating it for Capitalism*. Stanford, CA, Stanford University Press 2019.
- Crawford, Kate und Trevor Paglen: *Excavating AI – The Politics of Images in Machine Learning Training Sets*. In: *Excavating AI* (19.09.2019) Website. <https://www.excavating.ai> [abgerufen am 12.05.2020].
- Cunitz, Maria: *Urania propitia sive Tabulae Astronomicae mire faciles [...]*. Ölsnitz (Oelsna Silesiorum), Seyffert, Johann 1650.
- Darwen, Hugh: *The Relational Model – Beginning of an Era*. In: *IEEE Annals of the History of Computing* 34/4 (2012), S. 30–37, DOI: 10.1109/MAHC.2012.50.
- Darwen, Hugh: *Why there are no Relational Databases*. Manuskript (2015) <http://www.dcs.warwick.ac.uk/~hugh/TTM/Why-Are-There-No-Relational-DBMSs.pdf>.
- Date, Chris J.: *Codd's First Relational Papers – A critical Analysis*. Manuskript (2015) <http://www.dcs.warwick.ac.uk/~hugh/TTM/CJD-on-EFC%27s-First-Two-Papers.pdf> [abgerufen am 30.07.2017].
- Date, Chris J. und Hugh Darwen: *Foundation for Future Database Systems – The third Manifesto*. Reading, MA, Addison-Wesley Professional 2000.
- Date, Chris J.: *Date on Database – Writings 2000-2006*. New York, NY, Springer Science & Business Media 2006.
- Date, Chris J.: *Logic and Database – The Roots of Relational Theory*. Victoria, BC, Trafford Publ 2007.
- Date, Chris J.: *Oral History of C. J. Date*. Mountain View, California. Interviewer: Thomas Haigh, Computer History Museum, Mountain View, CA (13.06.2017).
- Date, Chris J.: *Relational Data Base Concepts*. In: *Datamation* (1976), S. 50–53.
- Dath, Dietmar: *Maschinenwinter – Wissen, Technik, Sozialismus, eine Streitschrift*. Frankfurt am Main, Suhrkamp 2008.
- Davenport, Tomas H.: *Holistic Management of Megapackage Change – The Case of SAP*. Bd. 153, 1996.
- Davies, Charles T.: *A Logical Concept for the Control and Management of Data*. Technical Report, AR-0803-00, IBM, Poughkeepsie (1967).
- Davies, Charles T.: *Data Processing Spheres of Control*. In: *IBM Systems Journal* 17/2 (1978), S. 179–198, DOI: 10.1147/sj.172.0179.
- Davies, Charles T.: *Recovery Semantics for a DB/DC System*. Proceedings of the ACM Annual Conference New York, NY, ACM 1973, S. 136–141, DOI: 10.1145/800192.805694.
- DB-Engines/Solid IT: *Ranking pro Datenbankmodell Kategorie*. In: *DB Engines* (03.2020) Website. [https://db-engines.com/de/ranking\\_categories](https://db-engines.com/de/ranking_categories) [abgerufen am 10.03.2020].
- De Landa, Manuel: *A new Philosophy of Society – Assemblage Theory and Social Complexity*. London, New York, NY, Continuum 2006.
- Deleuze, Gilles und Félix Guattari: *Tausend Plateaus*. Berlin, Merve-Verlag 1980.
- Deleuze, Gilles: *Postscript on the Societies of Control*. In: *October* 59 (1992), S. 3–7.

- Demuth, Birgit und Gesellschaft für Informatik (Hrsg.): Informatik in der DDR – Grundlagen und Anwendungen. Drittes Symposium Informatik in der DDR, 15 und 16 Mai 2008 in Dresden, Deutschland. Bonn, Köllen 2008.
- Demuth, Birgit und Karl-Heinz Wiggert: *Technologietransfer am Beispiel von DABA 1600*. In: Demuth, Birgit (Hrsg.): Informatik in der DDR – Grundlagen und Anwendungen. Bonn 2008, S. 280–291.
- Der Bundesbeauftragte für die Unterlagen der Stasi der DDR (Hrsg.): Hauptverwaltung A (HV A) – Aufgaben, Strukturen, Quellen. Berlin 2013.
- Der Leiter der Staatlichen Zentralverwaltung für Statistik: *Anordnung Nr 2 über die Einführung und Anwendung einheitlicher datenverarbeitungsgerechter Primärdokumente*. GBl DDR Teil 1, Nr. 6 (22.01.1974).
- Der Leiter der Staatlichen Zentralverwaltung für Statistik: *Anordnung Nr 3 über die Einführung und Anwendung einheitlicher datenverarbeitungsgerechter Primärdokumente*. GBl DDR Teil 1, Nr. 23 (26.07.1983).
- Der Leiter der Staatlichen Zentralverwaltung für Statistik: *Anordnung über die Einführung einheitlicher datenverarbeitungsgerechter Primärdokumente des einheitlichen Systems von Rechnungsführung und Statistik*. GBl DDR Teil 2, Nr. 118 (22.11.1968).
- Der Minister für Wissenschaft und Technik, Der Minister für Elektrotechnik und Elektronik und Der Leiter der Staatlichen Zentralverwaltung für Statistik: *Anordnung über die Informations- und Beratungsleistungen zur Entwicklung, Produktion und Mehrfachnutzung von Software in der DDR*. GBl Teil 1, Nr. 9 (14.02.1986).
- Der Staatssekretär für Forschung und Bildung: *Anordnung über die Nachnutzung wissenschaftlich-technischer Ergebnisse innerhalb der Deutschen Demokratischen Republik*. GBl DDR Teil 2, Nr. 32 (22.03.1967).
- Der Vorsitzende des Staatsrates der DDR: *Gesetz über das einheitliche Bildungssystem*. GBl DDR Teil 1, Nr. 6 (25.02.1965).
- DeSofi, Oliver J.: *Intrigue, Capitalism, Love My True Story*. Pittsburg, Red Lead Press 2009.
- Didi-Huberman, Georges und J. M. Charcot: *Invention of Hysteria – Charcot and the photographic Iconography of the Salpêtrière*. Cambridge, MA, MIT Press 1982.
- Diebold, John: *Automation – The Advent of the automatic Factory*. Princeton, N.J., van Nostrand 1952.
- Dieter, Michael, Carolin Gerlitz, Anne Helmond, u. a.: *Multi-Situated App Studies – Methods and Propositions*. In: *Social Media & Society* 5/2 (2019), DOI: 10.1177/2056305119846486.
- D’Ignazio, Catherine und Lauren F. Klein: *Data Feminism*. Cambridge, MA, The MIT Press 2020.
- Dollner, Reiner: *Einheitliche Primärdokumente rationalisieren Fertigungsorganisation*. In: *Rechentechnik/Datenverarbeitung* 18/12 (1981), S. 1.
- Dollner, Reiner und Lutz Weickert: *Einheitliche technologische Auftragsbelege – ein Mittel zur effektiven Anwendung der EDV für innerbetriebliche Rationalisierung*. In: *Rechentechnik/Datenverarbeitung* 10/2 (1973), S. 11–15.
- Domman, Monika: *Wertspeicher – Epistemologien des Warenlagers*. In: *Zeitschrift für Medien- und Kulturforschung* 2 (2012), S. 35–50, DOI: 10.28937/1000106367.
- Donig, Simon: *Die DDR-Computertechnik und das COCOM-Embargo 1958-1973 – Technologietransfer und institutioneller Wandel im Spannungsverhältnis zwischen Sicherheit und Modernisierung*. In: Schade, Gabriele und Friedrich Naumann (Hrsg.): *Informatik in der DDR – Eine Bilanz*. Bonn 2006, S. 251–272.
- Donig, Simon: *Vorbild und Klassenfeind – Die USA und die DDR-Informatik*. In: *Osteuropa* 59/10 (2009), S. 89–100.
- Dotzler, Bernhard J. und Henning Schmidgen (Hrsg.): *Parasiten und Sirenen – Zwischenräume als Orte der materiellen Wissensproduktion*. Bielefeld, transcript 2008.
- Driscoll, Kevin: *From Punched Cards to »Big Data« – A Social History of Database Populism*. In: *communication +1* 1/1 (2012), S. 1–33, DOI: 10.7275/R5B8562P.

- Dummer, G. W. A., F. P. Thomson und J. Mackenzie Robertson: *Banking Automation*. Oxford, NY, Pergamon Press 1971.
- Earle, Robert J.: *Method and Apparatus for Storing and Retrieving multi-dimensional Data in Computer Memory*. Patent, US5359724A (25.10.1994).
- Ebeling, Knut und Stephan Günzel (Hrsg.): *Archivologie – Theorien des Archivs in Wissenschaft, Medien und Künsten*. Berlin, Kulturverlag Kadmos 2009.
- Eck, Stefan: *Database Technology*. In: *We made it possible*. (11.2011) <http://www.wemadeitpossible.com/2011/01/database-technology/> [abgerufen am 25.11.2018].
- Elias, Norbert: *Über den Prozeß der Zivilisation*, Bd. 1. Frankfurt am Main, Suhrkamp 1939.
- Engemann, Christoph und Andreas Jahn-Sudmann (Hrsg.): *Machine Learning – Medien, Infrastrukturen und Technologien der künstlichen Intelligenz*. Bielefeld, Transcript 2018.
- Engemann, Christoph: *You cannot not Transact – Big Data und Transaktionalität*. *Big Data*. 2014, S. 365–381.
- Engster, Frank und Phoebe V Moore: *The Search for (Artificial) Intelligence in Capitalism*. In: *Capital & Class* 44/2 (2020), S. 201–218, DOI: 10.1177/0309816820902055.
- Eppelmann, Rainer, Horst Möller, Günter Nooke, u. a. (Hrsg.): *Lexikon des DDR-Sozialismus – Das Staats- und Gesellschaftssystem der Deutschen Demokratischen Republik*, Bd. 1. 2. Aufl. Stuttgart, UTB 1997.
- Epstein, Bob: *History of Sybase*. In: *IEEE Annals of the History of Computing* 35/2 (2013), S. 31–41, DOI: 10.1109/MAHC.2012.52.
- Eswaran, K. P., J. N. Gray, R. A. Lorie, u. a.: *The notions of consistency and predicate locks in a database system*. In: *Communications of the ACM* 19/11 (1976), S. 624–633, DOI: 10.1145/360363.360369.
- Eubanks, Virginia: *Automating Inequality – How High-Rech Tools profile, police, and punish the Poor*. 2019.
- Evans, Harold, Gail Buckland und David Lefer: *They made America – From the Steam Engine to the Search Engine, two Centuries of Innovators*. 1. Aufl. New York, NY, Little, Brown 2004.
- Evie Discusses My Baby Carrot. (20.09.2015) <https://www.youtube.com/watch?v=ax5HCzzlmFE>.
- Farnam, Henry: *Die innere französische Gewerbepolitik von Colbert bis Turgot*, Bd. 1. Leipzig, Duncker & Humblot 1878.
- Few, Stephen: *Show me the numbers*. Berkeley, CA, Analytics Press 2012.
- Fiedler, Reinhard und Helga Haubold: *Eine Datenbank zur Unterstützung der Analysentätigkeit*. In: *Rechentchnik/Datenverarbeitung* 13/4 (1976), S. 48–53.
- Fincham, Robin (Hrsg.): *Expertise and Innovation – Information Technology Strategies in the Financial Services Sector*. Oxford, New York, NY, Clarendon Press, Oxford University Press 1994.
- Fischer, Michael J.: *Luca Pacioli on business profits*. In: *Journal of Business Ethics* 25/4 (2000), S. 299–312, DOI: 10.1023/A:1006281415873.
- Fisher, Eran: *Audience Labour on Social Media – Learning from Sponsored Stories*. In: Fisher, Eran und Christian Fuchs (Hrsg.): *Reconsidering Value and Labour in the Digital Age* Palgrave Macmillan 2015, S. 115–132.
- Fisher, Eran und Christian Fuchs: *Reconsidering Value and Labour in the Digital Age*. Palgrave Macmillan 2015.
- Fiske, John: *Media matters – Race and Gender in US Politics*. Minneapolis, University of Minnesota Press 1996.
- Forsman, Sarah: *OLAP Council White Paper*. In: *The OLAP Council (1997) Website*. <http://www.olapcouncil.org/research/whtpaply.htm> [abgerufen am 19.05.2018].
- Foucault, Michel: *Archäologie des Wissens*. Frankfurt am Main, Suhrkamp 1969.
- Foucault, Michel: *Sicherheit, Territorium, Bevölkerung – Geschichte der Gouvernementalität I*, Bd. 1. Frankfurt am Main, Suhrkamp 1978.



- Fox, A. und E.A. Brewer: *Harvest, yield, and scalable tolerant Systems*. Proceedings of the Seventh Workshop on Hot Topics in Operating Systems Rio Rico, AZ, IEEE Comput. Soc 1999, S. 174–178, DOI: 10.1109/HOTOS.1999.798396.
- Frabetti, Federica: *Eine neue Betrachtung der Digital Humanities im Kontext originärer Technizität*. In: Reichert, Ramón (Hrsg.): *Big Data: Analysen zum digitalen Wandel von Wissen, Macht und Ökonomie*. Bielefeld, Transcript 2014, S. 85–102.
- Frankston, Bob: *Implementing Visicalc*. In: RMF (12.10.2015) Website. <http://rmf.vc/implementing-visicalc?pdf=t>.
- Frederik, Jesse und Maurits Martijn: *The new dot com bubble is here: it's called online advertising*. In: The Correspondent (06.11.2019) Online Magazin. <https://thecorrespondent.com/100/the-new-dot-com-bubble-is-here-its-called-online-advertising/13228924500-22d5fd24> [abgerufen am 18.11.2019].
- Fritsche, Detlev: *Mit Prototyprekonstruktion zum Welthöchststand? PC-Software in den letzten Jahren der DDR*. In: *Dresdener Beiträge zur Geschichte der Technikwissenschaften* 30 (2005), S. 105–123.
- Frühbrodt, Lutz: *Streit um Software-Erfolg: Lieferte Robotron die Vorlage für SAP-Software?* In: *Die Welt* (22.10.2007).
- Fry, James P. und Edgar H. Sibley: *Evolution of Data-Base Management Systems*. In: *ACM Computing Surveys* 8/1 (1976), S. 7–42, DOI: 10.1145/356662.356664.
- Fuchs, Christian: *Internet and Society – Social Theory in the Information Age*. New York, NY, Routledge 2008.
- Fuchs-Kittowski, Klaus, Horst Kaiser, Reiner Tschirschwitz, u. a.: *Informatik und Automatisierung*, Bd. 1, hrsg. v. Klaus Fuchs-Kittowski. Berlin, Akademie Verlag 1976.
- Fuchs-Kittowski, Klaus, Reiner Tschirschwitz und Bodo Wenzlaff: *Mensch und Automatisierung – Methodologische Probleme auf dem Weg zur dynamischen automatisierten Informationsverarbeitung*. In: *Deutsche Zeitschrift für Philosophie* 21/Sonderheft (1973), S. 104–121.
- Fuchs-Kittowski, Klaus: *Zur Herausbildung von Sichtweisen der Informatik in der DDR unter Einfluss der Kybernetik I und II Ordnung*. In: Dittmann, Frank und Rudolf Seising (Hrsg.): *Kybernetik steckt den Osten an – Aufstieg und Schwierigkeiten einer interdisziplinären Wissenschaft in der DDR*. Bd. 1, Berlin, Trafo 2007, S. 323–382.
- Fuller, Matthew und Andrew Goffey: *Evil Media*. Cambridge, MA, MIT Press 2012.
- Fuller, Matthew, Andrew Goffey, Olga Goriunova, u. a.: *Editorial 1 – A Billion Gadget Minds*. In: *Computational Culture* (11.2011) Online Journal. <http://computationalculture.net/editorial1/> [abgerufen am 12.10.2019].
- Galloway, Alexander: *Protocol – How Control exists after Decentralization*. Cambridge, MA, MIT Press 2004.
- Ge, Liping und Stefan Voß: *ERP Application in China – An Overview*. In: *International Journal of Production Economics* 122/1 (2009), S. 501–507, DOI: 10.1016/j.ijpe.2009.06.009.
- Geijsbeek, John B.: *Ancient double-entry Bookkeeping – Lucas Pacioli's Treatise*. Denver, Colorado 1914.
- Gerovitch, Slava: *From newspeak to cyberspeak – a history of Soviet cybernetics*. Cambridge, MA, MIT Press 2002.
- Gfh: *Die Software AG erwirbt das relationale DBMS von Nixdorf*. In: *Computerwoche* (11.01.1991).
- Giannetti, Claudia und Siegfried Zielinski (Hrsg.): *AnArchive(s) – Eine minimale Enzyklopädie zur Archäologie und Variantologie der Künste und Medien*. Köln, König 2014.
- Giebler, Horst: *Mikrorechnersystem Robotron K 1600*. In: *Neue Technik im Büro* 24/2 (1980), S. 39–40.
- Gießmann, Sebastian: *Elemente einer Praxistheorie der Medien*. In: *Zeitschrift für Medienwissenschaft* 19/19–2 (2018), S. 95–109, DOI: 10.14361/zfmw-2018-100212.
- Gießmann, Sebastian, Tobias Röhl und Sonja Trischer (Hrsg.): *Materialität der Kooperation*. Wiesbaden, Springer 2019.

- Gitelman, Lisa: Paper Knowledge – Towards a Media History of Documents. Duke University Press 2014.
- Gitelman, Lisa: Raw data is an Oxymoron. Cambridge, MA, MIT Press 2013.
- Glaisher, James: *The Mathematical Table*. Encyclopedia Britannica. Bd. 26, Cambridge, Cambridge University Press 1911, S. 325–336.
- Goebel, Jacqueline: *Deutsche Post kürzt Prognose – Das Computer-Chaos des Frank Appel*. In: Wirtschaftswoche Online (29.10.2015) Online Zeitung. <https://www.wiwo.de/unternehmen/dienstleister/deutsche-post-kuerzt-prognose-das-computer-chaos-des-frank-appel/12514056.html> [abgerufen am 05.11.2019].
- Goffman, Erving: *Frame Analysis – An Essay on the Organization of Experience*. Boston, Northeastern University Press 1974.
- Goody, Jack: *The Domestication of the Savage Mind*. Cambridge, MA, Cambridge University Press 1977.
- Gore, Albert Arnold „Al“: *Vice President Remarks*. Buenos Aires (21.03.1994) <https://search.itu.int/history/HistoryDigitalCollectionDocLibrary/4.144.57.en.104.pdf>.
- Gräßler, Rolf: *Die Entwicklung von Sachgebietsorientierten Programiersystemen SOPS des VEB Kombinat Robotron*. In: Schade, Gabriele und Friedrich Naumann (Hrsg.): *Informatik in der DDR – Eine Bilanz*. Bonn 2006, S. 116–139.
- Gräßler, Rolf: *Möglichkeiten der Rationalisierung der Einsatzvorbereitung elektronischer Datenverarbeitungsanlagen durch sachgebietsorientierte Programmiersysteme für die Probleme der Planung und Leitung von sozialistischen Industriebetrieben und Kombinat*. Dissertation, TU Dresden, Dresden (01.07.1972).
- Gräßler, Rolf: *Zielstellung und Arbeitsweise der Sachgebietsorientierten Informations- und Beratungseinrichtung des VEB Kombinat Robotron*. In: *Neue Technik im Büro* 31/3 (1987), S. 76–77.
- Grattan-Guinness, Igor: *The Computation Factory – de Prony's Project for making Tables in the 1790's*. In: Campell-Kelly et. al. (Hrsg.): *The history of mathematical tables – From Sumer to spreadsheets*. Oxford, NY, Oxford University Press 2003, S. 105–123.
- Grattan-Guinness, Igor und H. J. M. Bos (Hrsg.): *From the Calculus to Set Theory, 1630-1910 – an Introductory History*. Princeton, Princeton University Press 2000.
- Graunt, John: *Natural and Political Observations Made upon the Bills of Mortality*. London, Roycroft/Dicas 1662.
- Gray, Jim: *An Interview with Jim Gray*. Interviewer: Philip L. Frana, Charles Babbage Institute Center for the History of Information Processing University of Minnesota, Minneapolis (03.01.2002).
- Gray, Jim (Hrsg.): *The Benchmark Handbook – For Database and Transaction Processing Systems*. San Mateo, CA, M. Kaufmann Publishers 1991.
- Grier, David Alan: *Table Making for the Relief of Labour*. In: Campell-Kelly et. al. (Hrsg.): *The History of Mathematical Tables – From Sumer to Spreadsheets*. Oxford, NY, Oxford University Press 2003, S. 256–292.
- Grier, David Alan: *The Relational Database and the Concept of the Information System*. In: *IEEE Annals of the History of Computing* 34/4 (2012), S. 9–17, DOI: 10.1109/MAHC.2012.70.
- Grier, David Alan: *When Computers were Human*. Princeton, N.J.; Woodstock, Princeton University Press 2005.
- Griswold, William G.: *Program Restructuring as an Aid to Software Maintenance*. Dissertation, University of Washington (1991) <http://cseweb.ucsd.edu/~wgg/Abstracts/gristhesis.pdf>.
- Grunert, Bodo: *Erprobung von BASTEI im 1 Moskauer Uhrenwerk S M Kirow*. In: *Rechentechnik/Datenverarbeitung* 12/3 (1975), S. 32–33.
- Gugerli, David: *The World as Database – On the Relation of Software Development, Query Methods and Interpretative Independence*. In: *Information & Culture* 47/3 (2012), S. 287–311, DOI: 10.7560/IC47302.

- Gumbrecht, Hans Ulrich: *Mediengeschichte als Wahrheitsereignis – Zur Singularität von Friedrich A. Kittlers Werk*. Die Wahrheit der technischen Welt. Berlin, Suhrkamp 2013, S. 396–421.
- Günther, Manfred: *Gemeinsame Arbeit am ESER wird ausgebaut und weiterentwickelt*. In: Rechen-technik/Datenverarbeitung 11/10 (1974), S. 2–5.
- Gutmann, Gernot: *Planungsmechanismen der Wirtschaft*. In: Eppelmann, Reiner (Hrsg.): Lexikon des DDR-Sozialismus. Paderborn, München, Wien, Ferdinand Schöningh 1996, S. 469–471.
- Haderle, Donald J. und Cynthia M. Saracco: *The History and Growth of IBM's DB2*. In: IEEE Annals of the History of Computing 35/2 (2013), S. 54–66, DOI: 10.1109/MAHC.2012.55.
- Haerder, Andreas und Theo Reuter: *Principles of Transaction-Oriented Database Recovery*. In: Computing Surveys 15/4 (1983), DOI: 10.1145/289.291.
- Haigh, Thomas: *A veritable Bucket of Facts – Origins of the Data Base Management System*. In: ACM SIGMOD Record 35/2 (2006), S. 33–49, DOI: 10.1145/1147376.1147382.
- Haigh, Thomas: *Fifty Years of Databases*. In: ACM Sigmod (11.12.2012) Blog. <http://wp.sigmod.org/?p=688> [abgerufen am 25.03.2013].
- Haigh, Thomas: *Finding a Story for the History of Computing*. 3, Collaborative Research Center 1187 Media of Cooperation. University Siegen, Siegen (07.2018).
- Haigh, Thomas: *How Data got its Base – Information Storage Software in the 1950s and 1960s*. In: IEEE Annals of the History of Computing 31/4 (2009), S. 6–25, DOI: 10.1109/MAHC.2009.123.
- Haigh, Thomas: *Inventing Information Systems – The Systems Men and the Computer, 1950-1968*. In: The Business History Review 75/1 (2001), S. 15–61, DOI: 10.2307/3116556.
- Haigh, Thomas und Mark Priestley: *Innovators assemble – Ada Lovelace, Walter Isaacson, and the Superheroines of Computing*. In: Communications of the ACM 58/9 (2015), S. 20–27, DOI: 10.1145/2804228.
- Haigh, Thomas: *Software in the 1960s as Concept, Service, and Product*. In: IEEE Annals of the History of Computing 24/1 (2002), S. 5–13, DOI: 10.1109/85.988574.
- Halpern, Orit: *Beautiful Data – A history of Vision and Reason since 1945*. Durham, Duke University Press 2014.
- Hammer, J., H. Garcia-Molina, J. Widom, u. a.: *The Stanford Data Warehousing Project*. Technical Report, 1995–10, Stanford Infolab (1995).
- Hänseroth, Thomas: *Eine Gründungsschrift der Technikwissenschaftsgeschichte in Deutschland – Kommentar zu Gisela Buchheim: Zur Wechselwirkung von Naturwissenschaften und Technikwissenschaften in ihrer historischen Entwicklung (1978)*. In: NTM Zeitschrift für Geschichte der Wissenschaften, Technik und Medizin 18/3 (2010), S. 401–408.
- Härder, Theo: *Implementierung von Datenbanksystemen*. München, Wien, Hanser 1978.
- Hartmann, Arendt, Hans-Dieter, Peter Naumann, Rolf-Wenzel Wenzel, u. a.: *Das Robotron Informationsrecherchesystem AIDOS – Software-Technologie von gestern?* In: Demuth, Birgit (Hrsg.): *Informatik in der DDR – Grundlagen und Anwendungen*. Bonn 2008, S. 280–291.
- Hartmann, Frank: *Die Logik der Datenbank Zwischen Leibnitz und Google – Otlet der Weltbibliothekar*. In: Hartmann, Frank (Hrsg.): *Vom Buch zur Datenbank. Paul Otlets Utopie der Wissensvisualisierung*. Berlin, Avinus 2012.
- Hartmann, Frank: *Materialität und Visibilisierung von Information – Zum Design des Interface*. In: Heibach, Christiane und Carsten Rohde (Hrsg.): *Ästhetik der Materialität*. München, Fink 2015, S. 97–121.
- Hartmann, Hans-Dieter: *AIDOS/M16-Informationsrecherchesystem*. In: *Rechentchnik/Datenverarbeitung* 24/3 (1987), S. 32–36.
- Hartmann, Jens: *Grenzgänger: Das Gehirn von Robotron*. In: *Die Welt* (10.08.2009).
- Harvey, Adam und Julien LaPlace: *MegaPixels – Origins and Endpoints of Biometric Datasets »In the Wild«*. (2019) Website. <https://megapixels.cc> [abgerufen am 05.09.2020].

- Hashagen, Ulf: *Rechner für die Wissenschaft – »Scientific Computing« und Informatik im deutschen Wissenschaftssystem 1870–1970*. In: Hashagen, Ulf und Hans Dieter Hellige (Hrsg.): *Rechnende Maschinen im Wandel – Mathematik, Technik, Gesellschaft*. Deutsches Museum 2011, S. 111–152.
- Haupt, Herman: *General Theory of Bridge Construction*. New York, NY, D. Appleton & Company 1873.
- Hawthorn, Paula: Oral-History: Paula Hawthorn. Interviewer: Janet Abbate (05.07.2002).
- Hayles, Katherine: *How we became Posthuman – Virtual Bodies in Cybernetics, Literature, and Informatics*. Chicago, Ill., University of Chicago Press 1999.
- Head, R.V.: *Getting Sabre off the Ground*. In: *IEEE Annals of the History of Computing* 24/4 (2002), S. 32–39, DOI: 10.1109/MAHC.2002.1114868.
- Heilmann, Till A.: *Datenarbeit im »Capture«-Kapitalismus – Zur Ausweitung der Verwertungszone im Zeitalter informatischer Überwachung*. (2015), DOI: 10.25969/mediarep/1561.
- Heilmann, Till A.: *Textverarbeitung – eine Mediengeschichte des Computers als Schreibmaschine*. Bielefeld, transcript Verlag 2012.
- Heilmann, Till A.: *Zur Vorgängigkeit der Operationskette in der Medienwissenschaft und bei Leroi-Gourhan*. In: *Internationales Jahrbuch für Medienphilosophie* 2/1 (2016), DOI: 10.1515/jbmp-2016-0103.
- Heinemann, Rolf: *Anwendung und weitere Entwicklung von Datenbankbetriebssystemen des VEB Kombinat Robotron*. In: *Neue Technik im Büro* 27/6 (1983), S. 162–163.
- Heinemann, Rolf: *Datenbanken als Rationalisierungsmittel in der Projektierung – Ergebnisse und Tendenzen*. In: *Neue Technik im Büro* 29/4 (1985), S. 108–109.
- Heinrich, Michael: *Die Wissenschaft vom Wert – Die Marxsche Kritik der politischen Ökonomie zwischen wissenschaftlicher Revolution und klassischer Tradition*. Münster, Verl. Westfälisches Dampfboot 1999.
- Heinrich, Michael: *Kritik der politischen Ökonomie – Eine Einführung*. Stuttgart, Schmetterling-Verl. 2005.
- Hempel, Ursula und Hans Loley: *Das relationale Datenbankbetriebssystem REDABAS*. In: *Rechen-technik/Datenverarbeitung* 22/11 (1985), S. 32–34.
- Hempel, Ursula und Hans Loley: *Datenbanken mit Personalcomputern*. 2. Aufl. Berlin, Verl. Die Wirtschaft 1988.
- Hempel, Ursula und Hans Loley: *Datenverwaltung mit REDABAS-3*. Berlin, Verl. Die Wirtschaft 1989.
- Henslee, Bill: *From BOMP to SaaS and Beyond – 1960s: BOMP & the Early Days of MRP*. In: e2btek.com (13.10.2016) <http://e2btek.com/bomp-saas-beyond-1960s/> [abgerufen am 09.03.2019].
- Herrmann, Jeffrey W.: *A History of Production Scheduling*. In: Herrmann, Jeffrey W. (Hrsg.): *Handbook of production scheduling*. New York, NY, Springer 2006.
- Hertle, Hans-Hermann: *Die Diskussion der ökonomischen Krisen in der Führungsspitze der SED*. In: Pirker, Theo und Mario Rainer Lepsius (Hrsg.): *Der Plan als Befehl und Fiktion – Wirtschaftsführung in der DDR. Gespräche und Analysen*. Opladen, Westdt. Verlag 1995, S. 309–346.
- Hill, Malcom R.: *West-East Technology Transfer – The British Experience*. In: Oda, Hiroshi (Hrsg.): *Law and politics of West-East technology transfer*. Dordrecht/Boston/Norwell, MA, M. Nijhoff/Graham & Trotman 1991.
- Hitomi, Katsundo: *Manufacturing Systems Engineering*. 2. Aufl. London ; Bristol, PA, Taylor and Francis 1996.
- Hoffedank, Günther und Bodo Grunert: *Einheitliche Primärdokumente als Bindeglied zwischen ökonomischen Prozess und maschineller Datenverarbeitung*. In: *Rechentchnik/Datenverarbeitung* 10/3 (1973), S. 22–28.
- Hörl, Erich: *Die Ökologisierung des Denkens*. In: *Zeitschrift für Medienwissenschaft* 14 (2016), S. 33–45, DOI: 10.25969/mediarep/1713.

- Horn, D., Jürgen Stumpf und F. Ziegenbalg: *Datenorganisation für das Mikrorechnersystem Robotron K 1600*. In: Neue Technik im Büro 25/1 (1981), S. 1–3.
- Hounshell, David A.: *From the American System to Mass Production, 1800–1932. The Development of Manufacturing Technology in the United States*. Baltimore, Hopkins Univ. Press 1984.
- Hübner, Peter: *Menschen-Macht-Maschinen Technokratie in der DDR*. In: Zentrum Für Zeithistorische Forschung (Hrsg.): *Eliten im Sozialismus. Beiträge zur Sozialgeschichte der DDR*. Köln, Böhlau 1999, S. 325–360.
- Hui, Yuk: *On the Existence of Digital Objects*. Minneapolis, University of Minnesota Press 2016.
- Hui, Yuk: *Recursivity and Contingency*. London, New York, NY, Rowman & Littlefield International 2019.
- Hunger, Francis: *Deep Love Algorithm Exhibition and Database Derive in Berlin*. In: databasecultures (07.02.2015) <http://databasecultures.irmielin.org/deep-love-algorithm-in-berlin/> [abgerufen am 27.09.2019].
- Hunger, Francis: *Epistemic Harvest – The electronic database as discourse and means of data production*. In: A Peer Reviewed Journal About Research Values 7/1 (2018), DOI: 10.7146/aprja.v7i1.115064.
- Hunger, Francis: *How to Hack Artificial Intelligence*. In: Colakides, Yiannis, Inte Gloerich und Marc Garrett (Hrsg.): *State Machines – Reflections and Actions at the Edge of Digital Citizenship, Finance, and Art* Amsterdam, Institute of Network Cultures 2019, S. 129–144.
- Hunger, Francis (Hrsg.): *Setun – Eine Recherche über den sowjetischen Ternärcomputer / Setun – An Inquiry into the Soviet Ternary Computer*. Leipzig, Inst. für Buchkunst 2007.
- Hunt, Edwin S.: *The medieval super-companies – A study of Peruzzi Company of Florence*. Cambridge, Cambridge University Press 1994.
- Hürtgen, Renate: *Angestellt im VEB – Loyalitäten, Machtressourcen und soziale Lagen der Industrieangestellten in der DDR*. Münster, Westfälisches Dampfboot 2009.
- IBM: *Customer Information Control System (CICS) – General Information Manual*. White Plains, NY 1972.
- IBM: *DB2 OLAP Server – theory and practices*. San Jose, Calif, IBM Corp., International Technical Support Organization 2001.
- IBM: *IBM 305 RAMAC – Manual of Operation*. New York, NY, IBM 1957.
- IBM (Hrsg.): *Manufacturing Accounting and Production Information Control System (MAPICS) – Introducing Order Processing and Accounting Applications*. Atlanta, GA 1983.
- IBM: *SHARE 7090 9PAC Part1*. White Plains, NY 1961.
- IBM SPREAD Task Group: *Processor Products Final Report of SPREAD task group*. (28.12.1961).
- Ingold, Tim: *Beyond Art and Technology – The Anthropology of Skill*. In: Schiffer, Michael B. (Hrsg.): *Anthropological Perspectives on Technology*. Albuquerque, University of New Mexico Press 2001, S. 17–31.
- Inmon, William H.: *Building the Data Warehouse*. 3. Aufl. New York, NY, J. Wiley 1992.
- Jenning, Matthias: *Die EDV in der Materialwirtschaft*. In: *Rechentechnik/Datenverarbeitung* 16/11 (1979), S. 30–34.
- Jensen, Sören und Ursula Schwarzer: *Anachronistischer Einkauf*. In: *Manager Magazin* (04.11.2013) Online Magazin. <https://www.manager-magazin.de/magazin/artikel/a-891269.html> [abgerufen am 05.11.2019].
- Jeschke, Herbert und Redaktion: *ESER-Test für problemorientierte Systemunterlagen*. In: *Rechentechnik/Datenverarbeitung* 11/6 (1974), S. 1.
- Jochum, Georg und Simon Schapp: *Die Steuerungswende Zur Möglichkeit einer nachhaltigen und demokratischen Wirtschaftsplanung im digitalen Zeitalter*. Marx und die Roboter. Vernetzte Produktion, Künstliche Intelligenz und lebendige Arbeit. Berlin, Dietz Verlag 2019, S. 327–344.

- Jonas, Eric und Konrad Paul Kording: *Could a Neuroscientist Understand a Microprocessor?* In: PLOS Computational Biology 13/1 (2017), DOI: 10.1371/journal.pcbi.1005268.
- Jungnickel, Hans-Georg: *15 Jahre erfolgreiche Beteiligung der DDR am Einheitlichen System Elektronischer Rechenmaschinen (ESER)*. In: Rechentechnik/Datenverarbeitung 21/10 (1984), S. 5–8.
- Jungnickel, Hans-Georg: *Rechentechnik der DDR im ESER – Beiträge zur DDR-Geschichte des Einheits-System der Elektronischen Rechentechnik (1968-1990)*. In: ESER DDR (2013) Website. <http://www.eser-ddr.de/> [abgerufen am 22.01.2016].
- Kahn, M.A., D.L. Rumelhart und B.L. Bronson: MICRO Information Management System Reference Manual. Ann Arbor, MI, University of Michigan/Institute of Labor and Industrial Relations 1970.
- Kaiser, Monika: *Machtwechsel von Ulbricht zu Honecker - Funktionsmechanismen der SED-Diktatur in Konfliktsituationen 1962 bis 1972*. Berlin, Akademie Verlag 1997.
- Känel, Siegfried von, Hans-Walter Thiele und Werner Jöstel: *Komplexe Betriebsplanung einer rechnerintegrierten Produktion*. In: Rechentechnik/Datenverarbeitung 26/2 (1989), S. 15–16.
- Kannenbergh, Axel: *»Elwis« ist tot – Lidl stoppt millionenschweres Projekt mit SAP*. In: Heise online (16.07.2018) Online Magazin. <https://www.heise.de/newsticker/meldung/Elwis-ist-tot-Lidl-stoppt-millionenschweres-Projekt-mit-SAP-4111245.html> [abgerufen am 05.11.2019].
- Keller, Bernd, Michael Starruß und Wolfgang Belke: *Konzept einer COBOL-orientierten Datenmanipulationssprache REDAMS für Kleinrechner*. Informationsreihe der TU Dresden, Dresden (04.08.1978).
- Keller, Bernd und Michael Starruß: *Dissertation: Entwurf und experimentelle Implementierung der COBOL-orientierten relationalen Datenmanipulationssprache REDAMS für Mikrorechner*. In: Rechentechnik/Datenverarbeitung 18/9 (1981), S. 25.
- Keller, Bernd. und Michael. Starruß: *Entwurf und experimentelle Implementierung der COBOL-orientierten relationalen Datenmanipulationssprache REDAMS für Kleinrechner*. Dissertation, TU Dresden, Dresden (17.12.1980).
- Kidwell, Peggy A.: *Stalking the Elusive Computer Bug*. In: IEEE Annals of the History of Computing 20/4 (1998), S. 5–9, DOI: 10.1109/85.728224.
- Kindleberger, Charles P.: *Commercial Expansion and the Industrial Revolution*. In: Journal of European Economic History 4/3 (1975), S. 613–654.
- Kitov, Anatolij I.: *Программирование экономических и управленческих задач*. Советское радио (Sovjetskoje Radio) 1971.
- Kittler, Friedrich A.: *Aufschreibesysteme 1800–1900*. 4. Aufl. München, Fink 1982.
- Kittler, Friedrich A.: *Der Kopf schrumpft – Herren und Knechte im Cyberspace*. In: Frankfurter Allgemeine Zeitung (10.03.1995), S. 29.
- Kittler, Friedrich A.: *Die Stadt ist ein Medium*. Die Wahrheit der technischen Welt – Essays zur Genealogie der Gegenwart. Frankfurt am Main, Suhrkamp 1988, S. 181–211.
- Klaus, Georg: *Kybernetik und Gesellschaft*. Berlin, Deutscher Verlag der Wissenschaften 1964.
- Kohli, Manu: *Using Machine Learning Algorithms on data residing in SAP ERP Application to predict equipment failures*. In: International Journal of Engineering & Technology 7/2.28 (2018), S. 312, DOI: 10.14419/ijet.v7i2.28.12952.
- Körner, Joachim: *Softwareentwicklung in Chemnitz*. Manuskript (2012) [https://eser-ddr.de/documents/Softwareentwicklung\\_AKoe.pdf](https://eser-ddr.de/documents/Softwareentwicklung_AKoe.pdf).
- Kosche, Wolfram: *Entscheidungskriterien für die Nutzung des DBS/R*. In: Rechentechnik/Datenverarbeitung 17/12 (1980), S. 13–15.
- Kosel, Gerhard: *Produktivkraft Wissenschaft*. Berlin, Verlag Die Wirtschaft 1957.
- Kosslyn: *Understanding Charts and Graphs*. In: Applied Cognitive Psychology 21/3 (1989), S. 185–225, DOI: 10.1002/acp.2350030302.
- Krajewski, Markus: *In Formation – Aufstieg und Fall der Tabelle als Paradigma der Datenverarbeitung*. Nach Feierabend: Zürcher Jahrbuch für Wissensgeschichte – Datenbanken. Diaphanes 2007, S. 37–55.

- Krajewski, Markus: *Paper Machines – about Cards & Catalogs, 1548-1929*. Cambridge, MA, MIT Press 2011.
- Krämer, Sybille und Christine Brandner: *Die Schrift als Instrument der Erkenntnis – Sybille Krämer im Gespräch mit Christine Brandner*. In: ALL-OVER. Magazin für Kunst und Ästhetik. 5 (2013).
- Krämer, Sybille und Horst Bredekamp: *Kultur, Technik, Kulturtechnik – Wider die Diskursivierung der Kultur*. In: Krämer, Sybille und Horst Bredekamp (Hrsg.): *Bild, Schrift, Zahl*. München, Fink 2003, S. 11–22.
- Krämer, Sybille: *Das Auge des Denkens – Visuelle Epistemologie am Beispiel der Diagrammatik. Fünfte Vorlesung: Descartes – Die Erkenntniskraft des Graphischen. Methodische Implikationen der Cartesischen Koordinatengeometrie*. VL5 Manuskript (2010.2009) <http://userpage.fu-berlin.de/~sybkram/media/downloads/VL5 - Descartes - Die Erkenntniskraft des Graphischen.pdf>.
- Krämer, Sybille: *Notationen Schemata und Diagramme – »Räumlichkeit« als Darstellungsprinzip*. In: Brandstetter, Gabriele, Franck Hofmann und Kirsten Maar (Hrsg.): *Notationen und choreographisches Denken*. Freiburg/Berlin/Wien, Rombach Verlag 2010a, S. 29–45.
- Krämer, Sybille: *Roots and Media of Computational Power – Some Remarks on the Genesis and Genius of Quantification in Early European Modernity*. In: Gramelsberger, Gabriele (Hrsg.): *From Science to Computational Sciences*. Zürich, Diaphanes 2011, S. 47–56.
- Krämer, Sybille: *Symbolische Maschinen*. Darmstadt, Wissenschaftliche Buchgesellschaft 1988.
- Krämer, Sybille: *Zwischen Anschauung und Denken – Zur epistemologischen Bedeutung des Graphismus*. In: Bromand, Joachim (Hrsg.): *Was sich nicht sagen lässt. Das Nicht-Begriffliche in Wissenschaft, Kunst und Religion*. Berlin, Akademie Verlag 2010b, S. 173–192.
- Krasin, Ivan, Tom Duerig, Neil Alldrin, u. a.: *OpenImages – A Public Dataset for Large-Scale Multi-Label and Multi-Class Image Classification*.
- Kromann, Frank M.: *Beginning PHP and MySQL*. New York, NY, Springer Science & Business Media 2018.
- Kruczynski, Dr.: *Untersuchungen zum REDABAS-JOIN*. In: *Rechentechnik/Datenverarbeitung* 24/11 (1987), S. 33–34.
- Kuipers, Giseline: *Communicative Figurations – Towards a New Paradigm for the Media Age?* In: Hepp, Andreas, Andreas Breiter und Uwe Hasebrink (Hrsg.): Cham, Springer 2018, S. 425–436.
- Kunau, Hans: *EDV – Instrument und Objekt der Intensivierung*. In: *Rechentechnik/Datenverarbeitung* 13/2 (1976), S. 1–2.
- Kutschera, Franz von: *Gottlob Frege – Eine Einführung in sein Werk*. Berlin, New York, NY, De Gruyter Studienbuch 1989.
- Laitko, Hubert: *Produktivkraft Wissenschaft, wissenschaftlich-technische Revolution und wissenschaftliches Erkennen – Diskurse im Vorfeld der Wissenschaftswissenschaft*. In: Rauh, Hans-Christoph und Peter Ruben (Hrsg.): *Denkversuche – DDR-Philosophie in den 60er Jahren* 1. Aufl. Berlin, Ch. Links 2005, S. 459–540.
- Lane, Robert E.: *The Decline of Politics and Ideology in a Knowledgeable Society*. In: *American Sociological Review* 31/5 (1966), S. 649, DOI: 10.2307/2091856.
- Latour, Bruno: *Drawing Things Together – Die Macht der unveränderlich mobilen Elemente*. ANThology – ein einführendes Handbuch zur Akteur-Netzwerk-Theorie. Bielefeld, transcript Verlag 2006, S. 259–307.
- Latour, Bruno: *Eine neue Soziologie für eine neue Gesellschaft: Einführung in die Akteur-Netzwerk-Theorie*, übers. v. Gustav Roßler. Frankfurt am Main, Suhrkamp 2010.
- Latour, Bruno: *Science in Action – How to follow Scientists and Engineers through Society*. Cambridge, MA, Harvard University Press 1987.
- Latour, Bruno: *We have never been modern*. Cambridge, MA, Harvard University Press 1991.
- Lauer, Josh: *Plastic surveillance – Payment cards and the history of transactional data, 1888 to present*. In: *Big Data & Society* (2020), S. 1–14, DOI: 10.1177/2053951720907632.

- Lauwers, Luc und Marleen Willekens: *Five hundred Years of Bookkeeping – a Portrait of Luca Pacioli*. In: Tijdschrift voor Economie en Management 39/3 (1994), S. 289–304.
- Lavoie, Don: *Rivalry and Central Planning – The Socialist Calculation Debate reconsidered*. Cambridge, NY, Cambridge University Press 1985.
- Lazzarato, Maurizio: *Immaterielle Arbeit – Ästhetisierung der Politik und der Produktion unter den Bedingungen des Postfordismus*. In: Atzert, Thomas (Hrsg.): *Umherschweifende Produzenten – Immaterielle Arbeit und Subversion*. Berlin, ID Verlag 1998.
- Lehmann, Nikolaus Joachim: *Zur Geschichte des »Instituts für maschinelle Rechentechnik« der Technischen Hochschule/ Technischen Universität Dresden*. Zur Geschichte von Rechentechnik und Datenverarbeitung in der DDR 1946-1968. Dresden, Hannah-Arendt-Institut für Totalitarismusforschung 1996, S. 123–157.
- Leibniz, Gottfried Wilhelm: *Entwurff gewisser Staats-Tafeln*. Sämtliche Schriften und Briefe. Bd. 3, Berlin, Akademie Verlag 1680, S. 340–349.
- Leimbach, Timo: *Die Geschichte der Softwarebranche in Deutschland – Entwicklung und Anwendung von Informations- und Kommunikationstechnologie zwischen den 1950ern und heute*. Dissertation, Fakultät für Geschichts- und Kunstwissenschaften. TU München, München (28.01.2009).
- Leimbach, Timo: *Vom Programmierbüro zum globalen Softwareproduzenten*. In: Zeitschrift für Unternehmensgeschichte 52/1 (2007).
- Lepsius, Mario Rainer: *Handlungsspielräume und Rationalitätskriterien der Wirtschaftsfunktionäre in der Ära Honecker*. In: Pirker, Theo und Mario Rainer Lepsius (Hrsg.): *Der Plan als Befehl und Fiktion – Wirtschaftsführung in der DDR. Gespräche und Analysen*. Opladen, Westdt. Verlag 1995, S. 347–362.
- Leroi-Gourhan, André: *Gesture and Speech*. Cambridge, MA, MIT Press 1964.
- Levien, R. E. und M. E. Maron: *A Computer System for Inference Execution and Data Retrieval*. In: Communications of the ACM 10/11 (1967), S. 715–721, DOI: 10.1145/363790.363817.
- Leybourne, William: *The Ready Reckoner*. 21. Aufl. London, Law and Son 1789.
- Li, Fei-Fei, Jia Deng, Wei Dong, u. a.: *ImageNet: A Large-Scale Hierarchical Image Database*. 2009 IEEE Conference on Computer Vision and Pattern Recognition Miami, FL, IEEE 2009, S. 248–255, DOI: 10.1109/CVPR.2009.5206848.
- Liu, Alan: *Transcendental Data – Toward a Cultural History and Aesthetics of the New Encoded Discourse*. In: Critical Inquiry 31/1 (2004), S. 49–84, DOI: 10.1086/427302.
- Livingston, Paul M: *The Politics of Logic – Badiou, Wittgenstein, and the Consequences of Formalism*. London, New York, NY, Routledge 2011.
- Lodahl, Hans-Jürgen: *VEB Robotron Projekt Dresden – Ein Betrieb des Kombinat VEB Robotron*. Manuskript, Dresden (30.11.2005).
- Loll, Franz, Richard Kurth, Bernhard Malsch, u. a.: *Datenbankerfahrungen*. In: Rechentechnik/Datenverarbeitung 19/7 (1982), S. 18–21.
- Loll, Franz: *Vorhandenes wirksamer nutzen*. In: Rechentechnik/Datenverarbeitung 19/4 (1982), S. 36.
- Lorenz, Hans-Günther: *Konferenz über AIDOS*. In: Rechentechnik/Datenverarbeitung 26/7 (1989), S. 34.
- Lovink, Geert: *Networks without a Cause – A Critique of Social Media*. Cambridge, UK, Malden, MA, Polity 2011.
- Luhmann, Niklas: *Die Gesellschaft der Gesellschaft*, Bd. 1. Frankfurt am Main, Suhrkamp 1998.
- Luhmann, Niklas: *Soziale Systeme – Grundriß einer allgemeinen Theorie*. 4. Aufl. Frankfurt am Main, Suhrkamp 1987.
- Lupton, Deborah: *Digital Companion Species and eating Data – Implications for theorising Digital Data-Human Assemblages*. In: Big Data & Society 3/1 (2016), DOI: 10.1177/2053951715619947.
- Liotard, Jean-François und Fredric Jameson: *The postmodern Condition – A Report on Knowledge*, übers. v. Geoff Bennington und Brian Massumi. Manchester, Manchester University Press 1984.



- Mackenzie, Adrian: *Machine learners – Archaeology of a Data Practice*. Cambridge, MA, The MIT Press 2017.
- Mackenzie, Adrian: *More Parts than Elements – How Databases multiply*. In: *Environment and Planning D: Society and Space* 30/2 (2010), S. 335–350, DOI: 10.1068/d6710.
- Mackinnon, Nick: *The Portrait of Fra Luca Pacioli*. In: *The Mathematical Gazette* 77/479 (1993), S. 130, DOI: 10.2307/3619717.
- Manas, Miroslav: *Operational Research in the Czech Republic*. In: *European Journal of Operational Research*. 87 (1995), S. 430–434, DOI: 10.1016/0377-2217(95)00213-8.
- Manovich, Lev: *Software takes Command – Extending the Language of New Media*. London, New York, NY, Bloomsbury 2013.
- Manovich, Lev: *The Language of New Media*. Cambridge, MA, MIT Press 2001.
- Mansoux, Aymeric und Roel Roscam Abbing: *Seven Theses on the Fediverse and the Becoming of FLOSS*. In: Gansing, Kristoffer und Inga Luchs (Hrsg.): *Eternal Network – The Ends and Becomings of Network Culture*. Amsterdam, Institut of Network Culture 2020, S. 124–140.
- Maron, M.E.: *An Historical Note on the Origins of Probabilistic Indexing*. In: *Information Processing & Management* 44/2 (2008), S. 971–972, DOI: 10.1016/j.ipm.2007.02.012.
- Marx, Karl: *Das Kapital – Kritik der politischen Ökonomie*, Bd. 23. Berlin, Dietz Verlag 1890.
- Marx, Karl: *Grundrisse der Kritik der politischen Ökonomie*, Bd. 42. Berlin, Dietz Verlag 1858.
- Mattern, Shannon: *Databodies in Codespace*. In: *Places Journal* 2018 (2018), DOI: 10.22269/180417.
- McCarthy, John: *A Basis for a Mathematical Theory of Computation*. *Studies in Logic and the Foundations of Mathematics*. Bd. 35, Elsevier 1961, S. 33–70.
- McGee, W. C.: *Data Base Technology*. In: *IBM Journal of Research and Development* 25/5 (1981), S. 505–519, DOI: 10.1147/rd.255.0505.
- McGoveran, David: *Observations on the 1969 Relational Operations*. Manuskript, Felton/CA (2014/2005) <http://www.alternativetech.com/publications/Observations%20on%20the%201969%20Relational%20Operators%20-%20edited%202014.pdf> [abgerufen am 31.08.2017].
- McJones, Paul: *The 1995 SQL Reunion – People, Projects, and Politics*.
- McLeod, Dennis und Monte Meldman: *RISS – A generalized Minicomputer Relational Data Base Management System*. *Proceedings of the May 19-22, 1975 AFIPS '75 conference Anaheim, California*, ACM Press 1975, S. 397.
- McLuhan, Marshall: *Die Gutenberg-Galaxis – Das Ende des Buchzeitalters*. Bonn, Don Mills, ON, Addison-Wesley 1962.
- Mealy, George H.: *Another Look at Data*. *International Workshop on Managing Requirements Knowledge* (1967), Anaheim (14.11.1967).
- Medina, Eden: *Cybernetic Revolutionaries – Technology and Politics in Allende's Chile*. Cambridge, MA, MIT Press 2014.
- Meier, Herrmann, Dietmar Hopp und Hasso Plattner: *Auftragssteuerung, Disposition, und Versandsteuerung integriert im Realzeitbetrieb*. Stuttgart, IBM Deutschland 1972.
- Meissner, Gerd: *SAP, die heimliche Software-Macht – Wie ein mittelständisches Unternehmen den Weltmarkt eroberte*. München, Heyne 1997.
- Mendelsohn, Andrew: *The Oracle Story: 1984–2001*. In: *IEEE Annals of the History of Computing* (2013), S. 10–23, DOI: 10.1109/MAHC.2012.56.
- Menhart, Franz und Jürgen Stumpf: *Das Datenbankbetriebssystem DABA 1600 Version 20*. In: *Rechentechnik/Datenverarbeitung* 24/2 (1987a), S. 16–18.
- Menhart, Franz und Jürgen Stumpf: *SQL 1630 - Datenbanksprache von DABA 1600*. In: *Rechentechnik/Datenverarbeitung* 24/9 (1987b), S. 15–18.
- Menhart, Franz und Jürgen Stumpf: *Wertung des relationalen Datenbankbetriebssystems DABA 1600*. In: *Rechentechnik/Datenverarbeitung* 24/4 (1987c), S. 5–8.

- Menhart, Franz und Jürgen Stumpf: *Zum Entwurf des Datenbankbetriebssystems DABA 1600*. In: *Rechentechnik/Datenverarbeitung* 24/7 (1987d), S. 9–12.
- Menninger, Karl: *Number Words and Number Symbols – A Cultural History of Numbers*. New York, NY, Dover Publications 1898.
- Merkel, Gerhard: *Computerentwicklungen in der DDR – Rahmenbedingungen und Ergebnisse*. In: Naumann, Friedrich und Gabriele Schade (Hrsg.): *Informatik in der DDR – eine Bilanz, Symposien, 7 bis 9 Oktober 2004 in Chemnitz, 11 bis 12 Mai 2006 in Erfurt* Bonn, Ges. für Informatik 2006, S. 40–54.
- Merkel, Gerhard, Siegfried Junge und et. al.: *Sammlung von Beiträgen zur Geschichte der Zentralen Forschungs- und Entwicklungseinrichtung des VEB Kombinat Robotron*. Manuskript, Dresden (15.02.2006).
- Merkel, Gerhard: *VEB Kombinat Robotron, Sitz Dresden – Ein Kombinat des Ministeriums für Elektrotechnik und Elektronik der DDR*. Manuskript, Dresden (30.12.2005).
- Mersch, Dieter: *Visuelle Argumente – Zur Rolle der Bilder in den Naturwissenschaften*. In: Maasen, Sabine, Torsten Mayerhauser und Cornelia Renggli (Hrsg.): *Bilder als Diskurse, Bilddiskurse*. Weilerswist, Velbrück Wissenschaft 2006, S. 95–116.
- Meyer, Heinrich: *Uebersicht der Geschichte der Kunst bei den Griechen*. 5. Aufl. Dresden, Walthersche Buchhandlung 1810.
- Meyer, Joseph: *Meyers Großes Konversations-Lexikon – Ein Nachschlagewerk des allgemeinen Wissens (1902–1908)*. Bd. 19, 6. Aufl. Leipzig/Wien, Bibliographisches Institut 1908.
- Meyer, Stephen: *The Five Dollar Day – Labor Management and Social Control in the Ford Motor Company, 1908–1921*. Albany, NY, State University of New York Press 1981.
- Meynen, Gloria: Büro. Dissertation, Humboldt-Universität zu Berlin, Berlin (2004).
- Michaels, Ann S., Benjamin Mittman und C. Robert Carlson: *A Comparison of the Relational and CODASYL Approaches to Data-Base Management*. In: *ACM Computing Surveys* 8/1 (1976), S. 125–151, DOI: 10.1145/356662.356668.
- Minard, Philippe: *La Fortune du Colbertisme – Etat et Industrie dans la France des Lumières*. Paris, Fayard 1998.
- Ministerrat der DDR: *Beschluß zur Erhöhung der Effektivität und zur Durchsetzung der sozialistischen Rationalisierung bei der Einsatzvorbereitung für die elektronische Datenverarbeitung*. GBl DDR Teil 2, Nr. 60 (14.07.1971).
- Moore, Phoebe V.: *The mirror for (artificial) intelligence – In whose reflection?* In: *SSRN Electronic Journal* [pre-print] (2019), DOI: 10.2139/ssrn.3423704.
- Mormann, Hannah: *Das Projekt SAP – Zur Organisationssoziologie betriebswirtschaftlicher Standardsoftware*. Bielefeld, Transcript 2016.
- Mormann, Hannah: *Zur informationstheoretischen und organisationstheoretischen Formalisierung von Organisation*. In: Passoth, Jan-Hendrik und Josef Wehner (Hrsg.): *Quoten, Kurven und Profile – Zur Vermessung der sozialen Welt*. Wiesbaden, Springer 2013, S. 69–88.
- Morozov, Evgeny: *Capitalism's New Clothes – Shoshana Zuboff's new Book on Surveillance Capitalism emphasizes the former at the expense of the latter*. In: *The Baffler* (02.04.2019a) Online Magazin. <https://thebaffler.com/latest/capitalisms-new-clothes-morozov> [abgerufen am 02.10.2019].
- Morozov, Evgeny: *Digital Socialism? The Calculation Debate in the Age of Big Data*. In: *New Left Review* 116/117 (2019b), S. 33–67.
- Morozov, Evgeny: *The Planning Machine*. In: *The New Yorker* (06.10.2014).
- Morozov, Evgeny: *To save everything, click here – The Folly of Technological Solutionism*. New York, NY, PublicAffairs 2013.
- Müggenburg, Jan: *Lebhafte Artefakte – Heinz von Foerster und die Maschinen des Biological Computer Laboratory*. Göttingen, Konstanz University Press 2018.

- Müller, Karl-Heinz: *6 Problemseminar Datenbanken*. In: Rechentechnik/Datenverarbeitung 21/5 (1984), S. 35.
- Müller, Karl-Heinz: *Problemseminar Datenbanken*. In: Rechentechnik/Datenverarbeitung 12/4 (1975), S. 44.
- Müller, Karl-Heinz: *Problemseminar Datenbanken*. In: Rechentechnik/Datenverarbeitung 14/5 (1977), S. 35.
- Müller, Karl-Heinz: *Problemseminar Datenbanken*. In: Rechentechnik/Datenverarbeitung 19/7 (1982), S. 6.
- Nadler, Steven M.: *The best of all possible Worlds – A Story of Philosophers, God, and Evil in the Age of Reason*. Princeton, Princeton University Press 2010.
- Nake, Frieder: *Informatik und die Maschinisierung von Kopfarbeit*. In: Coy, Wolfgang (Hrsg.): *Sichtweisen der Informatik*. Braunschweig, Vieweg 1992, S. 181–202.
- National Information Standards Organization: *Understanding Metadata*. Bethesda, MD, NISO Press 2004.
- Naumann, Friedrich und Gabriele Schade (Hrsg.): *Informatik in der DDR – eine Bilanz, Symposien, 7 bis 9 Oktober 2004 in Chemnitz, 11 bis 12 Mai 2006 in Erfurt*. Bonn, Ges. für Informatik 2006.
- Nendel, Karl und Ralf Pasch: *General der Mikroelektronik – Autobiographie*. Berlin, Edition Berolina 2018.
- Neubert, Christoph und Serjoscha Wiemer: *Rewriting the Matrix of Life – Biomedica Between Ecological Crisis and Playful Actions*. In: *communication +1* 3/1 (2014), S. 1–31, DOI: 10.7275/R50V89RS.
- Nigel, Richard Fitz: *The Dialogue of the Exchequer*. Oxford University Press 1180.
- Nixon, Brice: *The Exploitation of Audience Labour – A Missing Perspective on Communication and Capital in the Digital Era*. In: Fisher, Eran und Christian Fuchs (Hrsg.): *Reconsidering Value and Labour in the Digital Age* Palgrave Macmillan 2015.
- Noble, Safiya Umoja: *Algorithms of Oppression – How Search Engines reinforce Racism*. New York, New York University Press 2018.
- North, Ken: *Sets, Data Models and Data Independence*. In: Dr. Dobb's (03.10.2010) <https://web.archive.org/web/20121024064523/https://www.drdoobs.com/database/sets-data-models-and-data-independence/228700616> [abgerufen am 10.05.2016].
- Nyckel, Eva-Maria: »Digital Taylorism«? *Data Practices and Governance in the Enterprise Software Salesforce*. In: Weizenbaum Series (2020), DOI: 10/gg8sqk.
- Olle, William T.: *Das Codasyl-Datenbankmodell*, übers. v. H. Münzenberger. Berlin, Heidelberg, New York, NY, Springer 1981.
- Orlicky, Joseph: *Material Requirements Planning – The New Way of Life in Production and Inventory Management*. New York, NY, McGraw-Hill 1974.
- Ott, Michaela: *Dividuationen: Theorien der Teilhabe*. Berlin, b\_books 2015.
- Ott, Michaela: *Es lebe die Dividuation! Zur Notwendigkeit anderer Denkkonzepte angesichts zeitgenössischer Teilhabepraktiken*. Manuskript, Berlin (2016).
- Palermo, Frank P.: *A Data Base Search Problem*. In: Tou, Julius T. (Hrsg.): *Information Systems*. Boston, MA, Springer 1974, S. 67–101.
- Parikka, Jussi: *What is Media Archaeology?* Cambridge, Polity Press 2012.
- Parks, Lisa: *Spotting the Satellite Dish – Populist Approaches to Infrastructure*. In: HMKV (Hrsg.): *Satellite/Border/Footprint*. Dortmund 2010, S. 2–7.
- Pasquinelli, Matteo: *How a Machine Learns and Fails – A Grammar of Error for Artificial Intelligence*. In: *Spheres. Journal for Digital Cultures*. 5 (2019), S. 1–17.
- Peeters, Jos: *Early MRP Systems at Royal Philips Electronics in the 1960s and 1970s*. In: *IEEE Annals of the History of Computing* 31/2 (2009), S. 56–69, DOI: 10.1109/MAHC.2009.23.
- Peirce, Charles S.: *Mathematical Philosophy*, hrsg. v. Carolyn Eisele. The Hague, Mouton [u.a.] 1976.

- Pendse, Nigel: *The Origins of today's OLAP Products*. In: The Olap Report (23.08.2007) <https://web.archive.org/web/20071221044811/http://www.olapreport.com/origins.htm>.
- Peters, Benjamin: *How not to Network a Nation – The uneasy History of the Soviet Internet*. Cambridge, MA, MIT Press 2017.
- Petschel, Dorit (Hrsg.): *175 Jahre TU Dresden – Die Professoren der TU Dresden, 1828-2003*, Bd. 3. Köln, Böhlau 2003.
- Petty, Sir William: *Political Arithmetick*. 3. Aufl. London, Peacock/Phoenix 1672.
- Pfaffenholz, Hans-Joachim: *Kommentar zur Anordnung Nr 2 über die Einführung und Anwendung einheitlicher datenverarbeitungsgerechter Primärdokumente*. In: *Rechentechnik/Datenverarbeitung* 11/10 (1974), S. 34–35.
- Pflug, Christa: *Informationsfond in rechnergestützten Informationssystemen für die Leitung und Planung*. In: *Rechentechnik/Datenverarbeitung* 13/4 (1976), S. 35–39.
- Phillips, Leigh und Michal Rozworski: *The People's Republic of Walmart – How the World's biggest Corporations are laying the Foundation for Socialism*. London, New York, NY, Verso 2019.
- Phrygio, Paulus Constantinus: *Chronicum Regum Regnorumque Omnium Catalogum, Et Perpetuum ab exordio mundi temporum, seculorumq[ue] seriem complectens : ex optimis quibusq[ue] Hebraeis, Graecis & Latinis autoribus congestum*. Basel, Johann Hervagius 1543.
- Pias, Claus: *Computer Spiel Welten*. Dissertation, Bauhaus Universität Weimar, Weimar (2000).
- Pias, Claus: *Was waren Medien-Wissenschaften? Stichworte zu einer Standortbestimmung*. In: Pias, Claus (Hrsg.): *Was waren Medien?* Zürich, Diaphanes 2011, S. 7–30.
- Pickering, Andrew: *The Cybernetic Brain – Sketches of another Future*. Chicago, London, University of Chicago Press 2010.
- Pieper, Helmut und Gerhard Fritsch: *Zum volkswirtschaftlich gezielten Einsatz von EDVA bei der weiteren Intensivierung*. In: *Rechentechnik/Datenverarbeitung* 13/4 (1976), S. 1–2.
- Pinto, Ana Teixeira: *The Pigeon in the Machine – The Concept of Control in Behaviorism and Cybernetics*. In: Pasquinelli, Matteo (Hrsg.): *Alleys of your Mind – Augmented Intelligence and its Traumas*. Lüneburg, Meson Press 2015.
- Plattner, Hasso: *Oral History of Hasso Plattner*. Interviewer: John Hollar, Computer Museum Palo Alto (04.12.2012).
- Polanyi, Michael: *Personal Knowledge – Towards a post-critical Philosophy*. Chicago, Univ. of Chicago Press 1958.
- Pollock, Friedrich: *Automation – Materialien zur Beurteilung ihrer ökonomischen und sozialen Folgen*. Frankfurt Main, Europäische Verlagsanstalt 1956.
- Pollock, Neil und Robin Williams: *Software and Organisations – The Biography of the Enterprise-Wide System or How SAP Conquered the World*. London, Routledge 2009.
- Pommerin, Reiner (Hrsg.): *175 Jahre TU Dresden – Geschichte der TU Dresden 1828-2003*, Bd. 1. Köln, Böhlau 2003.
- Pompe, Hedwig und Leander Scholz (Hrsg.): *Archivprozesse – die Kommunikation der Aufbewahrung*. Köln, DuMont 2002.
- Poovey, Mary: *A History of the modern Fact – Problems of Knowledge in the Sciences of Wealth and Society*. Chicago, University of Chicago Press 1998.
- Poster, Mark: *The second Media Age*. Cambridge, UK, Cambridge, MA, Polity Press 1995.
- Postone, Moishe: *Zeit, Arbeit und gesellschaftliche Herrschaft*. Freiburg, Ca Ira 2003.
- Preger, R.: *The Oracle Story, Part 1: 1977-1986*. In: *IEEE Annals of the History of Computing* 34/4 (2012), S. 51–57, DOI: 10.1109/MAHC.2012.54.
- Preuß, Susanne und Stuttgart: *ERP-Software – »Das ist schlimmer als Brexit, Trump und Handelskrieg«*. In: *Frankfurter Allgemeine Zeitung* (07.10.2019).

- Pritchett, Dan: *BASE – An ACID alternative*. In: Queue 6/3 (2008), S. 48–55, DOI: 10.1145/1394127.1394128.
- Problembeschreibung – Sachgebietsorientiertes Programmiersystem BASTEI*.
- Pulla, Ralf: *Elektrotechnik und Informationstechnik*. In: Hänseroth, Thomas (Hrsg.): 175 Jahre TU Dresden – Wissenschaft und Technik. Studien zur Geschichte der TU Dresden. Bd. 2, Köln, Böhlau 2003.
- Puttrich, Gottfried und W. Rinn: *Das Betriebsgeschehen in 80 Spalten – Teil I: Was ist bei der Einführung der Lochkartentechnik in Industriebetrieben zu beachten?* In: Neue Technik im Büro 4 (1960), S. 94.
- Quine, Willard V.: *From a logical Point of view – Nine Logico-Philosophical Essays*. 2. Aufl. Harper and Row 1953.
- Radkau, Joachim: *Revoltierten die Produktivkräfte gegen den real existierenden Sozialismus – Technikhistorische Anmerkungen zum Zerfall der DDR*. In: 1999 – Zeitschrift für Sozialgeschichte des 20. und 21. Jahrhunderts 5/4 (1990), S. 13–42.
- Rattan, Dhavleesh, Rajesh Bhatia und Maninder Singh: *Software Clone Detection – A systematic Review*. In: Information and Software Technology 55/7 (2013), S. 1165–1199, DOI: 10.1016/j.infsof.2013.01.008.
- Raunig, Gerald: *Dividuum – Maschinischer Kapitalismus und molekulare Revolution*, Bd. 1. Wien, Transversal Texts 2015.
- RDBMS Workshop: Oracle. Interviewer: Burton Grad, Mountain View, CA (2007).
- Rechenwerk Halle: *Demontage einer EC1056 EDVA*. In: Rechenwerk Halle (2007) Website. <http://rechenwerk.halle.it/usr/digital-ag/projekte/andere/eser1/> [abgerufen am 01.03.2019].
- Rechenwerk Halle: *ESEER Wechselp Plattenspeicher WPS EC5066M*. In: Rechenwerk Halle (08.09.2011) Website. <http://rechenwerk.halle.it/usr/digital-ag/projekte/technisch/EC5066/> [abgerufen am 02.03.2019].
- Redaktion: *Zur Anwendung von Redabas in der DDR*. In: Rechentechnik/Datenverarbeitung 26/9 (1989), S. 1.
- Rekoff, M. G.: *On Reverse Engineering*. In: IEEE Transactions on Systems, Man, and Cybernetics SMC-15/2 (1985), S. 244–252, DOI: 10.1109/TSMC.1985.6313354.
- Richardson, Lewis Fry: *Weather Prediction by Numerical Process*. Cambridge, Cambridge University Press 1922.
- Riewendt, Peter: *DAFEMA – Datenbanksystem für die Fernverarbeitung*. In: Rechentechnik/Datenverarbeitung 17/12 (1980), S. 24–25.
- Robles, Gregorio und Jesús M. González-Barahona: *A Comprehensive Study of Software Forks – Dates, Reasons and Outcomes*. In: Hammouda, Imed, Björn Lundell, Tommi Mikkonen, u. a. (Hrsg.): *Open Source Systems – Long-Term Sustainability* Bd. 378, Berlin, Heidelberg, Springer 2012, S. 1–14.
- Robotrontechnik.de: *Betriebssystem SCP*. In: robotrontechnik.de (01.03.2019a) Website. <http://www.robotrontechnik.de/index.htm?/html/software/scp.htm> [abgerufen am 20.04.2019].
- Robotrontechnik.de: *Verkehrstechnik – Rechnergestützte Dispatcherzentrale (RDZ/RZV)*. In: robotrontechnik.de (01.03.2019b) Website. <http://www.robotrontechnik.de/index.htm?/html/arbeitsplaetze/verkehr.htm> [abgerufen am 05.06.2019].
- Robson, Eleanor: *Tables and Tabular Formatting in Sumer, Babylonia, Assyria, 2500 BCE–50CE*. The History of Mathematical Tables – From Sumer to Spreadsheets. Oxford, NY, Oxford University Press 2003, S. 19–48.
- Rossiter, Ned: *Software, Infrastructure, Labor – A Media Theory of Logistical Nightmares*. New York, NY, Routledge Taylor & Francis Group 2016.
- Rößler, Johannes: *Goethe und Johann Heinrich Meyers Tabelle zur antiken Kunstgeschichte (1826)*. Stil-Linien diagrammatischer Kunstgeschichte. Berlin, München, Deutscher Kunstverlag 2014, S. 112–131.
- Rowe, Lawrence A.: *History of the Ingres Corporation*. In: IEEE Annals of the History of Computing 34/4 (2012), S. 58–70, DOI: 10.1109/MAHC.2012.51.

- Rudder, Christian: *Dataclysm – Who we are when we think no one's looking*. New York, NY, Crown Publishers 2014.
- Ruhland, Jürgen, Wilfried Voss und Peter Schreiber: *DASTI-DAFEMA – Datenbanksystem technischer Informationen*. In: *Rechentechnik/Datenverarbeitung* 26/12 (1989), S. 10–13.
- Sabrow, Martin: *Zukunftspathos als Legitimationsressource – Zu Charakter und Wandel des Fortschrittsparadigmas in der DDR*. In: Haupt, Heinz-Gerhard, Jörg Requate und Maria Köhler-Baur (Hrsg.): *Aufbruch in die Zukunft. Die 1960er Jahre zwischen Planungseuphorie und kulturellem Wandel – DDR, ČSSR und Bundesrepublik Deutschland im internationalen Vergleich*. Weilerswist, Velbrück Wissenschaft 2004, S. 165–184.
- Sack, Kurt: *Was sind SOPS, Typenprojekte und Software?* In: *Rechentechnik/Datenverarbeitung* 11/7 (1974), S. 27–28.
- SAP UEC (Anonymous): *R/3 History in Screen Shots*. In: SAP User Experience Community (16.04.2004) Website. <https://experience.sap.com/archived/r3-history-in-screen-shots/> [abgerufen am 15.03.2019].
- Schabacher, Gabriele: *Lokalisierbarkeit – Materialität – Technik*. In: Andriopoulos, Stefan, Gabriele Schabacher und Eckhard Schumacher (Hrsg.): *Die Adresse des Mediums*. Köln, DuMont 2001, S. 19–24.
- Schabacher, Gabriele: *Medium Infrastruktur – Trajektorien soziotechnischer Netzwerke in der ANT*. In: *Zeitschrift für Medien- und Kulturforschung* (2013), S. 129–148, DOI: 10.28937/1000107542.
- Schabacher, Gabriele: *Waiting – Cultural Techniques, Media, and Infrastructures*. In: Dünne, Jörg, Kathrin Fehringer, Kristina Kuhn, u. a. (Hrsg.): *Cultural Techniques. Spaces, Texts, Collectives*. London, New York, NY, De Gruyter 2020.
- Schärli, Alain: *Compter du Bout des Doigts: Cailloux, Jetons et Bouliers – de Périclès à nos Jours*. Lausanne, Presses polytechn. et univ. Romandes 2006.
- Schildhauer, Kurt: *Einheitliche technologische Primärdokumente Entwicklung – Einführung – Durchsetzung*. In: *Rechentechnik/Datenverarbeitung* 18/12 (1981), S. 5–7.
- Schmidt, Erwin: *Zur Geschichte und Lehre der Informatik an der Technischen Universität Dresden bis zur Gründung des Informatik-Zentrums 1986*. In: Demuth, Birgit (Hrsg.): *Informatik in der DDR – Grundlagen und Anwendungen*. Bonn 2008, S. 142–153.
- Schmidt-Burkhardt, Astrid: *Die Kunst der Diagrammatik. Perspektiven eines neuen bildwissenschaftlichen Paradigmas*. Bielefeld, Transcript 2012.
- Schneider, Birgit: *Textiles Prozessieren – Eine Mediengeschichte der Lochkartenweberei*. Zürich, Diaphanes 2007.
- Schoppan, Wolfgang: *Zur Ökonomie der Nutzung von Standard-Anwendersoftware*. In: *Rechentechnik/Datenverarbeitung* 18/10 (1981), S. 4–8.
- Schreiter, Dieter, Erhard Stange und Hans-Georg Vater: *Datenbankbetriebssystem MIMER und Programmbanksystem MADRAS*. In: *Rechentechnik/Datenverarbeitung* 19/11 (1982), S. 10–11.
- Schreiter, Dieter, Uwe Wloka, Peter Dramm, u. a.: *Erfahrungen bei der Anwendung eines relationalen Datenbanksystems*. In: *Rechentechnik/Datenverarbeitung* 24/7 (1987), S. 24–26.
- Schröter, Jens: *Medien und Ökonomie – Eine Einführung*. Wiesbaden, Springer 2019.
- Schubert, Dietrich: *Das relationale Datenbankbetriebssystem DABA 1600 – ein weiteres Ergebnis der Kooperation von Robotron und TU Dresden*. In: *Neue Technik im Büro* 29/4 (1985), S. 110–112.
- Schubert, Dietrich: *Stand und Perspektiven der Entwicklung von Datenbankbetriebssystemen*. In: *edv-aspekte* 4 (1977), S. 2–10.
- Schuster, Steward und Tandem Computers: *Relational Data Base Management for On-Line Transaction Processing*. Technical Report, 81.5 PN87604, Tandem Computers (02.1981).
- Schüttpelz, Erhard: *Die medienanthropologische Kehre der Kulturtechniken*. In: Engell, Lorenz, Joseph Vogl und Bernhard Siegert (Hrsg.): *Archiv für Mediengeschichte. Themenschwerpunkt – Kulturgeschichte als Mediengeschichte (oder vice versa?)* Universitätsverlag Weimar 2006, S. 1–23.

- Schüttpelz, Erhard: *Infrastrukturelle Medien und öffentliche Medien*. In: *Media in Action* 0 (2016), S. 1–21.
- Schwarz, Benjamin, Saumya Debray und Gregory Andrews: *Disassembly of executable code revisited*. In: *Proc. IEEE 2002 Working Conference on Reverse Engineering (WCRE) IEEE Computer Society 2002*, S. 45–54, DOI: 10.1109/WCRE.2002.1173063.
- Scoville, James G.: *The Taylorization of Vladimir Ilich Lenin*. In: *Industrial Relations* 40/4 (2001), S. 620–626, DOI: 10.1111/0019-8676.00228.
- Segal, Jerome: *Kybernetik in der DDR – Begegnung mit der marxistischen Ideologie*. In: *Dresdner Beiträge zur Geschichte der Technik und der Technikwissenschaften* 27 (2001), S. 47–75.
- Segelken, Barbara: *Bilder des Staates – Kammer, Kasten und Tafel als Visualisierungen staatlicher Zusammenhänge*. Berlin, Akademie Verlag 2010.
- Seifert, Johanna: *Praktiken, Techniken, Operationen*. In: *Internationales Jahrbuch für Medienphilosophie* 5/1 (2019), S. 99–114, DOI: 10.1515/jbmp-2019-050109.
- Selinger, Patricia G., M. M. Astrahan, Don Chamberlin, u. a.: *Access Path Selection in a Relational Database Management System*. ACM Press 1979, DOI: 10.1145/582095.582099.
- Senge, Peter M.: *The fifth Discipline – The Art and Practice of the Learning Organization*. New York, NY, Doubleday/Currency 1990.
- Service Geographique de l'Armee (de Prony): *Tables des Logarithmes a huit Decimales*. Paris, Imprimerie Nationale 1889.
- Shannon, Claude E.: *A Mathematical Theory of Communication*. In: *Bell System Technical Journal* 27/4 (1948), S. 623–656, DOI: 10.1002/j.1538-7305.1948.tb00917.x.
- Shannon, Claude E.: *The Bandwagon*. In: *IRE Transactions on Information Theory* 2/1 (1956), S. 3–3, DOI: 10.1109/TIT.1956.1056774.
- Shi, Hang und Eugene Hoyt: *What's Wrong with ERP in China?* In: Kurosu, Masaaki (Hrsg.): *Human-Computer Interaction. Theory, Design, Development and Practice* Bd. 9731, Cham, Springer 2016, S. 98–104.
- Shin, Sun-Joo: *The iconic Logic of Peirce's Graphs*. Cambridge, MA, MIT Press 2002.
- Siegele, Ludwig und Joachim Zepelin: *Matrix der Welt – SAP und der neue globale Kapitalismus*. Frankfurt am Main, New York, NY, Campus Verlag 2009.
- Siegert, Bernhard: *Cultural Techniques – Grids, Filters, Doors, and other Articulations of the Real*, übers. v. Geoffrey Winthrop-Young. New York, NY, Fordham University Press 2015.
- Siegert, Bernhard und Joseph Vogl (Hrsg.): *Europa – Kultur der Sekretäre*. Zürich, Diaphanes 2003.
- Simon, Herbert A. und Allen Newell: *Heuristic Problem Solving – The Next Advance in Operations Research*. In: *Operations Research* 6/1 (1958), S. 1–10, DOI: 10.1287/opre.6.1.1.
- Skolem, Thoralf: *Two Remarks on Set Theory*. In: *Mathematica Scandinavica* 5 (1957), S. 40.
- Sleifer, Jaap: *Planning ahead and falling behind – The East German Economy in comparison with West Germany 1946-2002*. Berlin, Akademie Verlag 2006.
- Smith, Adam: *An Inquiry into the Nature and Causes of the The Wealth of Nations*. London, W. Strahan & T. Cadell 1776.
- Sobeslavsky, Erich: *Der schwierige Weg von der traditionellen Büromaschine zum Computer. Zur Geschichte von Rechentechnik und Datenverarbeitung in der DDR 1946–1968*. Dresden, Hannah-Arendt-Institut für Totalitarismusforschung 1996, S. 7–122.
- Soja, Edward W.: *Postmetropolis – Critical Studies of Cities and Regions*. Malden, MA, Blackwell Pub 2000.
- Soll, Jacob: *The Information Master – Jean-Baptiste Colbert's secret State Intelligence System*. Ann Arbor, University of Michigan 2009.
- Soll, Jacob: *The Reckoning – Financial Accountability and the Rise and Fall of Nations*. New York, NY, Basic Books 2014.

- Sombart, Werner: *Der Moderne Kapitalismus – Die Genesis des Kapitalismus*, Bd. 1. München, Duncker & Humblot 1902.
- Sonnemann, Rolf: *Das Konzept der Geschichte der Produktivkräfte in der DDR Geschichtswissenschaft*. In: *Dresdener Beiträge zur Geschichte der Technikwissenschaften* 24 (1996), S. 1–19.
- Sonnemann, Rolf (Hrsg.): *Geschichte der Technischen Universität Dresden 1828 – 1978*. Berlin, VEB Deutscher Verlag der Wissenschaften 1978.
- Sprott, David: *Enterprise Resource Planning – Componentizing the Enterprise Application Packages*. In: *Communications of the ACM* 43/4 (2000), S. 63–69, DOI: 10.1145/332051.332074.
- Star, Susan Leigh und Karen Ruhleder: *Steps toward an Ecology of Infrastructure – Design and Access for large Information Spaces*. In: *Information systems research* 7/1 (1996), S. 111–134, DOI: 10.1287/isre.7.1.111.
- Starke (Demuth), Birgit: *Übersetzung und Optimierung relationaler Anfragen am Beispiel der Datenbanksprache SQL*. Dissertation, TU Dresden, Dresden (15.09.1983).
- Steiner, André: *Kein freies Spiel der Kräfte!? – Das neue ökonomische System als Einheit von Plan und Markt*. In: Haupt, Heinz-Gerhard, Jörg Requate und Maria Köhler-Baur (Hrsg.): *Aufbruch in die Zukunft. Die 1960er Jahre zwischen Planungseuphorie und kulturellem Wandel – DDR, ČSSR und Bundesrepublik Deutschland im internationalen Vergleich*. Weilerswist, Velbrück Wissenschaft 2004, S. 43–64.
- Steiner, Benjamin: *Die Ordnung der Geschichte – Historische Tabellenwerke in der Frühen Neuzeit*. Köln, Böhlau 2008.
- Stenovec, Tim: *Google has gotten incredibly good at predicting Traffic – Here's how*. In: *Business Insider* (18.12.2015) Online Magazin. <https://www.businessinsider.com/how-google-maps-knows-about-traffic-2015-11> [abgerufen am 11.03.2020].
- Steyerl, Hito, Wendy Hui Kyong Chun, Florian Cramer, u. a.: *Pattern Discrimination*. Lüneburg, Meson Press 2018.
- Stonebraker, Michael, Gerald Held, Eugene Wong, u. a.: *The Design and Implementation of INGRES*. In: *ACM Transactions on Database Systems* 1/3 (1976), S. 189–222, DOI: 10.1145/320473.320476.
- Stonebraker, Michael R., Eugene Wong und Gerald L. Held: *INGRES – A Relational Data Base System*. ACM Press 1975, S. 409.
- Stonebraker, Michael: *Retrospection on a Database System (INGRES)*. In: *ACM Transactions on Database Systems* 5/2 (1980), S. 225–240, DOI: 10.1145/320141.320158.
- Stys, Thomas: *AIDOS OS/ES in der wissenschaftlich-technischen Information*. In: *Rechentchnik/Datenverarbeitung* 18/3 (1981), S. 29–33.
- Suppes, Patrick: *Axiomatic Set Theory*. Princeton, NJ., van Nostrand 1960.
- Swade, Doron: *The unerring Vertainty of Mechanical Agency – Machines and Table making in the Nineteenth Century*. In: Campell-Kelly et. al. (Hrsg.): *The History of Mathematical Tables – From Sumer to Spreadsheets*. Oxford, NY, Oxford University Press 2003, S. 145–174.
- Systemunterlagen-Dokumentation – Sachgebietsorientiertes Programmiersystem BASTEI*.
- Tandem Computers: *Tandem NonStop Transaction Processing*. In: *Computerworld* (07.11.1983).
- Tansley, A. G.: *The Use and Abuse of Vegetational Concepts and Terms*. In: *Ecology* 16/3 (1935), S. 284–307, DOI: 10.2307/1930070.
- Taylor, Frederick Winslow: *Shop Management*. London, New York, NY, Harper & Brothers 1903.
- Taylor, Frederick Winslow: *The principles of scientific management*. London, New York, NY, Harper & Brothers 1911.
- Taylor, Norm: *Once Upon a MAPICS*. In: *MAPICS The Magazine*. 1 (1987).
- Tencent: *Tencent - dictionary\_and\_semantic\_hierarchy.txt*. (16.10.2018) [https://github.com/Tencent/tencent-ml-images/blob/master/data/dictionary\\_and\\_semantic\\_hierarchy.txt](https://github.com/Tencent/tencent-ml-images/blob/master/data/dictionary_and_semantic_hierarchy.txt) [abgerufen am 29.09.2020].



- Terranova, Tiziana: *Free Labour – Producing Culture for the Digital Economy*. London 2003.
- Terry, D. B., M. M. Theimer, Karin Petersen, u. a.: *Managing Update Conflicts in Bayou, a Weakly Connected Replicated Storage System*. Proceedings of the fifteenth ACM Symposium on Operating Systems Principles - SOSP '95. Copper Mountain, CO, ACM Press 1995, S. 172–182, DOI: 10.1145/224056.224070.
- The National Archives: *Exchequer of Ireland at Work*.
- Thomas, Gene: *Eras of MRP to LEAN*. In: *MRP to Lean* (02.03.2008) <http://mrp-to-lean.blogspot.com/2007/08/executive-summary-rev.html> [abgerufen am 09.03.2019].
- Thylstrup, Nanna Bonde: *Data out of Place – Toxic Traces and the Politics of Recycling*. In: *Big Data & Society* 6/2 (2019), S. 1–9, DOI: 10.1177/2053951719875479.
- Trogemann, Georg, Alexander Y. Nitussov und Wolfgang Ernst (Hrsg.): *Computing in Russia – The History of Computer Devices and Information Technology Revealed*. 1. Aufl. Braunschweig, Vieweg 2001.
- Tufte, Edward R.: *Envisioning Information*. Cheshire, Graphics Press 1995.
- Tufte, Edward R.: *The Visual Display of Quantitative Information*. Cheshire, Graphics Press 1983.
- Tufte, Edward R.: *Visual Explanations – Images and Quantities, Evidence and Narrative*. Cheshire, Graphics Press 1997.
- Vahrenkamp, Richard: *Die erste Informationsexplosion Die Rolle der Lochkartentechnik bei der Büro-rationalisierung in Deutschland 1910 bis 1939*. In: *Technikgeschichte* 84/3 (2017), S. 209–242, DOI: 10.5771/0040-117X-2017-3-209.
- Van Deventer, John H.: *Planning Production for Profit*. New York, NY, The Engineering Magazine Company 1921.
- Van Kerckhove, Marleen: *COCOM Rules*. In: *Computer Law & Security Review* 6/5 (1991), S. 9–12, DOI: 10.1016/0267-3649(91)90146-M.
- VEB Datenverarbeitungszentrum Magdeburg (Hrsg.): *DAFEMA – Datenbanksystem mit Fernzugriff über Bildschirm im Teilhaberbetrieb*. VEB Datenverarbeitungszentrum Magdeburg 1987.
- VEB Robotron (Hrsg.): *DABA 1600 Datenbankbetriebssystem – Anwendungsbeschreibung, POS, K 1600*. Dresden, VEB Kombinat Robotron, 1981.
- VEB Robotron (Hrsg.): *DABA 1600 Datenbankbetriebssystem – Methodische Hinweise; POS, K 1600*. Dresden, VEB Kombinat Robotron, 1984a.
- VEB Robotron (Hrsg.): *DABA 1600 Datenbankbetriebssystem – Sprachbeschreibung; POS, K 1600*. Dresden, VEB Kombinat Robotron, 1984b.
- VEB Robotron (Hrsg.): *Programm- und Anwendungsbeschreibung SOPS AIDOS*. Dresden, VEB Kombinat Robotron 1974.
- Virno, Paolo: *Grammatik der Multitude – Öffentlichkeit, Intellekt und Arbeit als Lebensformen*, Bd. 4, übers. v. Klaus Neundlinger. Wien, Turia & Kant 2008.
- Vismann, Cornelia: *Files – Law and Media Technology*, übers. v. Geoffrey Winthrop-Young. Stanford, Calif, Stanford University Press 2008.
- Vogel, Friedrich, Joachim Gres, Andrea Lederer, u. a.: *Bericht des 1 Untersuchungsausschusses nach Artikel 44 des Grundgesetzes (Kommerzielle Koordinierung)*.
- Vogl, Joseph: *Kalkül und Leidenschaft – Poetik des ökonomischen Menschen*. München, Sequenzia Verlag 2002.
- Wade, Bradford W. und Don Chamberlin: *IBM Relational Database Systems – The Early Years*. In: *IEEE Annals of the History of Computing* 34/4 (2012), S. 38–48, DOI: 10.1109/MAHC.2012.48.
- Waldmann, L.: *Datenbanksystem mit Anschlußmöglichkeit für das programmierbare Bildschirmterminal robotron PBT 4000 (DAFEMA)*. In: *Neue Technik im Büro* 25/4 (1981), S. 102–103.
- Ware, Willis H.: *RAND and the Information Evolution – A History in Essays and Vignettes*. Santa Monica, CA, Rand Corp. 2008.

- Wedekind, Hartmut: Datenbanksysteme I, Bd. 1. Mannheim, Bibliograph. Inst 1974.
- Weingarten, Rüdiger: *Datenbanken*. In: Günther und Ludwig (Hrsg.): *Schrift und Schriftlichkeit*. Berlin, New York, NY, De Gruyter 1994, S. 158–170.
- Wenk, Vera: Untersuchungen zur effektiven Implementierung des Relationenkonzeptes am Beispiel relationalalgebraorientierter Anfragesprachen. Dissertation, TU Dresden, Dresden (12.11.1981).
- Wenk, Vera: Zur Implementierung des Relationenkonzeptes für ESER. Informationsreihe der TU Dresden, Dresden (01.08.1978).
- Westervelt, Frank, B.W. Arden, B. A. Galler, u. a.: *Program and Addressing Structure in a Time-Sharing Environment*. In: *Journal of the ACM* 13/1 (1966), S. 1–16.
- Westervelt, Frank: CONCOMP – Research in conversational Use of Computers. Final report. ORA Project 07449, University of Michigan, Michigan (12.1970).
- White, Hylton: *Materiality, Form, and Context: Marx contra Latour*. In: *Victorian Studies* 55/4 (2013), S. 667–682, DOI: 10.2979/victorianstudies.55.4.667.
- Widlock, Thomas: *Kulturtechniken – Ethnographisch fremd und anthropologisch fremd*. In: Kienlin, Tobias L. (Hrsg.): *Fremdheit – Perspektiven auf das Andere*. Bonn, Habelt 2015, S. 41–59.
- Wiegand, Wayne A. und Donald G. Davis (Hrsg.): *Encyclopedia of Library History*. New York, NY, Garland Publishing 1994.
- Wiener, Norbert: *Cybernetics, or, Control and communication in the animal and in the machine*. 2. Aufl. New York, NY, Wiley 1948.
- Wiggert, Karl-Heinz, G. Bulla und J. Ernst: *Problemorientierte Systemunterlagen für das Mikrorechnergerätesystem robotron K 1600 – Datenbankbetriebssystem DABA 1600*. In: *Neue Technik im Büro* 25/2 (1982), S. 168–171.
- Wiggert, Karl-Heinz: *Die Ausbaustufe 20 des Datenbankbetriebssystems DABA 1600*. In: *Neue Technik im Büro* 30/2 (1986), S. 58–60.
- Wildensee, Christoph: *Entwicklung eines Leitfadens zur systematischen Anbindung von SAP-R/2*. Diplom, AKAD-Fachhochschule Pinneberg, Pinneberg (1997).
- Wilke, Manfred und Klaus Schroeder: *Parteitage und -konferenzen der SED*. In: Eppelmann, Reiner (Hrsg.): *Lexikon des DDR-Sozialismus*. Paderborn, München, Wien, Ferdinand Schöningh 1996a, S. 461–463.
- Wilke, Manfred und Klaus Schroeder: *Politbüro des ZK der SED*. In: Eppelmann, Reiner (Hrsg.): *Lexikon des DDR-Sozialismus*. Paderborn, München, Wien, Ferdinand Schöningh 1996b, S. 469–471.
- Wilke, Manfred und Klaus Schroeder: *Zentralkomitee (ZK) der SED*. In: Eppelmann, Reiner (Hrsg.): *Lexikon des DDR-Sozialismus*. Paderborn, München, Wien, Ferdinand Schöningh 1996c, S. 717–720.
- Williams, Bruce und Roger Johnson: *The Ready Reckoner*. In: *IEEE Annals of the History of Computing* 27/4 (2005), S. 64–80, DOI: 10.1109/MAHC.2005.64.
- Wilson, James M.: *Gantt charts – A centenary Appreciation*. In: *European Journal of Operational Research* 149/2 (2003), S. 430–437, DOI: 10.1016/S0377-2217(02)00769-5.
- Wilson, James M.: *The Origin of Material Requirements Planning in Frederick W Taylor's Planning Office*. In: *International Journal of Production Research* 54/5 (2016), S. 1535–1553, DOI: 10.1080/00207543.2015.1092616.
- Wilson, Mike: *The difference between God and Larry Ellison [inside Oracle Corporation]*. New York, NY, William Morrow and Company 1997.
- Winkler, Hartmut: *Prozessieren*. Lüneburg & YouTube (07.07.2015) <https://youtu.be/qygOLmqUSGA?t=3049>.
- Wylie, John: *Non-Representational Subjects?* In: Harrison, Paul und Ben Anderson (Hrsg.): *Taking-Place – Non-Representational Theories and Geography*. Routledge 2016, S. 99–115.
- Yannakoudakis, E. J.: *The architectural Logic of Database Systems*. London, New York, NY, Springer 1988.

- Yates, JoAnne: *Control through Communication – The Rise of System in American Management*. Baltimore, Md., Johns Hopkins Univ. Press 1993.
- Yates, JoAnne: *Structuring the Information Age – Life Insurance and Technology in the Twentieth Century*. Baltimore, Md, Johns Hopkins University Press 2005.
- Yelavich, Bob: *A Brief History of CICS*. In: Follow the Yelavich Road (1990) Website. [https://web.archive.org/web/20040306192750fw\\_/http://yelavich.com:80/cicshist.htm](https://web.archive.org/web/20040306192750fw_/http://yelavich.com:80/cicshist.htm).
- Zedler, Johann Heinrich: *Tabellen – Gedächtnis-Tabelle*. Universal-Lexicon aller Wissenschaften und Künste. Bd. 41, Leipzig, Halle 1744, S. 1286–1296.
- Zielinski, Siegfried: *Archäologie der Medien – Zur Tiefenzeit des technischen Hörens und Sehens*. Reinbek/Hamburg, Rowohlt 2002.
- Zielinski, Siegfried: *Deep Time of the Media – Toward An Archaeology of Hearing and Seeing by Technical Means*. Cambridge, MA, MIT Press 2006.
- Zillmann, Günter: *Beitrag der EDV zur weiteren Entwicklung der Volkswirtschaft der DDR*. In: *Rechentechnik/Datenverarbeitung* 16/9 (1979), S. 5–8.
- Zuboff, Shoshana: *Big Other – Surveillance Capitalism and the Prospects of an Information Civilization*. In: *Journal of Information Technology* 30/1 (2015), S. 75–89, DOI: 10.1057/jit.2015.5.
- Zuboff, Shoshana: *The Age of Surveillance Capitalism – The Fight for a Human Future at the new Frontier of Power*. New York, NY, PublicAffairs 2019.
- Zygiaris, Sotirios: *Database management systems – a business-oriented approach using ORACLE, MySQL and MS Access*. Bingley, UK, Emerald Publishing Limited 2018.
- Берг, А. И., А. И. Китов und А. А. Ляпунов: *О возможностях автоматизации управления народным хозяйством*. Vortrag, Moskau (11.1959) [http://www.computer-museum.ru/books/kitov\\_asu.htm](http://www.computer-museum.ru/books/kitov_asu.htm) [abgerufen am 15.04.2019].
- Центр государственного хранения космической документации: *ESER EC-1022 Großcomputer 1982*.

# Abbildungsverzeichnis

Abb. 1: Die Mehrzahl der weltweit eingesetzten Datenbankmanagementsysteme sind derzeit relational organisiert. (Grafik: Autor in Anlehnung an DB-Engines/Solid IT 2020).	4
Abb. 2: Edgar F. Codd auf dem ACM SIGFIDET Workshop 1974 (Foto: Ben Shneiderman, IBM Archives Poughkeepsie).	36
Abb. 3: Lochkartenleser (Punch Brushes) und Stapel (Stacker) (IBM 1957, 127).	39
Abb. 4: Schematischer Aufbau der RAMAC Festplatte (IBM 1957, 12).	39
Abb. 5: Login zum Michigan Terminal System auf einem IBM 3277 Display-Terminal (CC-BY 3.0 Michigan Terminal System Archive).	51
Abb. 6: Michigan Terminal System (CC-BY 3.0 Michigan Terminal System Archive).	52
Abb. 7: Aufbau von Domänen und Relationen im hierarchischen Modell (Codd 1970, 381).	61
Abb. 8: Diagrammatische Abbildung einer Relation vierten Grades in Form einer Tabelle (Codd 1970, 381).	63
Abb. 9: Die Schlüssel unterscheiden die Tupel (Autor).	65
Abb. 10: Verschachtelte Relationen (Autor).	65
Abb. 11: Die Normalisierung und Aufgliederung von sich wiederholenden Attributen in jeweils eigene Relationen (Autor).	65
Abb. 12: Die Relation ›ship‹ (Codd 1969, 6).	69
Abb. 13: Die Relation R und S (Fig. 4) werden mit einem natürlichen Join verbunden (Codd 1969, 6).	70
Abb. 14: Exchequer of Ireland (The National Archives 1450).	79
Abb. 15: Neperianische Rechen-Stäblein, hier in einer flachen Ausführung (CC BY-SA 3.0 Bernd Gross).	81
Abb. 16: Links: Willkürlich umgebrochener Fließtext im Vergleich zur Tabelle (Autor)	82
Abb. 17: Sortierung und Gruppierung als Operation (Autor).	85
Abb. 18: Vertikale Tabelle (Phrygio 1543, 284–285).	89
Abb. 19: Einzüge, gesperrte und kursivierte Schrift sowie verschiedenen Schriftgrößen dienen zur Strukturierung des Inhalts innerhalb der Zellen (Bredow 1810, 16–17, Ausschnitt).	94
Abb. 20: Die Zeilen erhalten eine durchgehende Zeilenhöhe, wodurch die Vergleichbarkeit zwischen den Spalten verstärkt wird. Leerstellen werden deutlicher sichtbar (Meyer/Heinrich 1810, 7–8).	95
Abb. 21: Tabelle aus dem Abregé des Königs von 1680. ([1680] Colbert 1865, II/2:783).	99
Abb. 22: Dispatch Board (Van Deventer 1921, 235).	102
Abb. 23: Produktionslaufkarte (Van Deventer 1921, 233ff.).	104
Abb. 24: Jahreszinsen zu 5 Prozent (Leybourne 1789, 224f.).	107
Abb. 25: Einfache Logarithmen mit fünf Dezimalstellen (De Prony 1889)	110
Abb. 26: Rechner bei Carl-Zeiss Jena (ZEISS Archiv, BACZ 10948)	112
Abb. 27: Tabelle Primärschablonen von Parabolspiegeln (Sonnefeld 1928b, ZEISS Archiv, BACZ 19839).	113
Abb. 28: Beispiel für das Erstellen einer Tabelle in SQL (Autor).	119
Abb. 29: Konstruktion der Leere (Carroll 1874).	120
Abb. 30: Auszug aus der Wordpress-Tabelle wp_posts (Screenshot aus Sequel Pro, Autor, 22.8.2019).	121
Abb. 31: Mengentheoretisch begründete Relation in Codd 1970(Codd/Edgar F. 1970).	124
Abb. 32: Jacopo de Barbari: Ritratto di Fra Luca Pacioli, ca. 1495 (Barbari 1495).	131
Abb. 33: Prinzip der doppelten Buchhaltung. (Autor).	132
Abb. 34: Double-entry ledger from J. and H. Hadden and Co. Ltd., Hosiers, Nottingham 1685-1915 (University of Nottingham, Ha A 3-5).	135
Abb. 35: Staatsstatistik: Einnahmen des Königs aus Gütern, Rechten und Veräußerungen in den Jahren bis 1680, zusammengestellt durch Colbert im Abrégé des Königs (Colbert 1865, II/2:771).	137
Abb. 36: Wissens-Arbeiterinnen an den Karteikartenschränken des Universellen Bibliographischen Verzeichnis des Mundaneums (Mundaneum, Archive Centre of the French Community of Wallonia-Brussels).	141

Abb. 37: Arbeitssaal der Computer Tabulating Company, Washington 1905 (Harris & Ewing Collection, Library of Congress, LC-H25- 91219-KG [P&P]).	141
Abb. 38: Büro einer Versicherungsfirma mit Women of Color als Büroarbeiterinnen, Chicago, April 1941 (Lee, Russell/Library of Congress, Prints & Photographs Division, FSA/OWI Collection, LC-DIG-fsa-8a29933).	142
Abb. 39: Dateneingabe bei der National Girobank, Bootle/ Lancashire ca. 1969 (Standard Telephones and Cables Ltd. In: (Dummer/Thomson/ Robertson 1971, 919ff.)).	142
Abb. 40: United Airlines Reservierungstafel, ca. 1930er Jahre (Smithsonian National Air and Space Museum Washington, 2000-9721)	144
Abb. 41: SABRE Buchung (IBM Archive, 1962).	146
Abb. 42: Beispiel für ein Ausgabeschema des Listengenerators 9PAC (IBM 1961, 11).	150
Abb. 43: Das IBM Terminal 2260 (Ryerson University – Library Archives, RG 63.72).	151
Abb. 44: Stückliste für ein Element des Cove Run Viaducts, Pennsylvania, (Haupt 1873, 195).	152
Abb. 45: Technische Zeichnung des Cove Run Viaducts mit Bezug auf die Stückliste (Haupt 1873, 271).	153
Abb. 46: Verschiedene Abschnitte einer Sphere of Control (Davies 1973, 137).	155
Abb. 47: Select-Abfragen ohne Verwendung eines Index; Zeit in Sekunden (Bitton/DeWitt/Turbyfill 1983, 15).	161
Abb. 48: Beispiel für die Werbung mit Fokus auf Transaktionen. (Tandem Computers 1983, 34–35).	162
Abb. 49: Schema für ein Data Warehouse (Hammer u. a. 1995).	165
Abb. 50: Illustration von OLAP-Würfeln in der Patentschrift für Essbase von Arbor Software Corp (Earle 1994).	166
Abb. 51: In der DDR verfügbare Datenbanksysteme (Autor).	187
Abb. 52: ESER-Bedienkonsole und Bandlaufwerk in der Sammlung Rechenwerk Halle/Saale (Autor)	193
Abb. 53: <i>System Data Base</i> – Cover der IBM Broschüre <i>Communication-Oriented Production Information Control System</i> , (IBM 1972a).	197
Abb. 54: IBM Copic Software Module (Hitomi 1996, 447).	198
Abb. 55: MAPICS Installationsvarianten (IBM (Hrsg.) 1983, Abschn 1–11, 1–12).	201
Abb. 56: Gegenstandskartei in Tabellenform (Breuning/Hoffmann 1973, 30).	203
Abb. 57: Fertigungsbeleg 3.3 (Büttner/Dollner/Kunitz 1981, 23).	211
Abb. 58: Übersicht Sachgebietsorientierte Programmiersysteme (Autorenkollektiv/Gräßler/Freudenberg 1975).	214
Abb. 59: Produktbaum in einem gerichteten Graphen (Ahrens 1976, 52).	220
Abb. 60: Überblicksschema BASTEI Datenverarbeitung mit den PLUS-Teilmodulen (Apitz/Saffian/Koslowski 1976, 30).	221
Abb. 61: Importierter ESER-Großcomputer aus der UdSSR (Wolfgang Thieme, Allgemeiner Deutscher Nachrichtendienst – Zentralbild, BARCH Bild 183-1983-1117-027).	223
Abb. 62: Eingabeformular mit Bestelldaten eines Auftrages (Meier/Hopp/Plattner 1972, 8).	
Abb. 63: Datenbank der Imperial Chemical Industries-Anwendung (Meier/Hopp/Plattner 1972, 8).	226
Abb. 64: Stand der Softwarefiguration SAP R/2 mit zehn Modulen Mitte der 1980er Jahre (SAP GmbH 1985b, 2)	229
Abb. 65: Personenstammdaten in SAP R/2 aus dem Jahr 1997 (SAP UEC (Anonymous) 2004).	232
Abb. 66: »Gliederung der geistig-produktiven Tätigkeiten unter informationellem Aspekt« (Fuchs-Kittowski u. a. 1976, 1:248), ursprünglich in (Fuchs-Kittowski/Tschirschwitz/Wenzlaff 1973, 118).	248
Abb. 67: Informationsfonds nach Christa Pflug (Pflug 1976, 37, 39).	249
Abb. 68: K 1600 Computer in einer sowjetischen Forschungseinrichtung, Mitte der 1980er Jahre in Kirow (Alexander Sacharnych/sovietcpu.com, o.J., privat)	259
Abb. 69: Datenbanksysteme des ESER 1977 laut Schubert (Schubert 1977, 6).	267
Abb. 70: »Architektur eines relationalen Datenbanksystems in topographischer Sicht«, Administratoren und Endnutzer greifen auf verschiedene topologische Aspekte des DBMS zu (Wenk 1978, 3).	268
Abb. 71: Cover der Datamation Ausgabe <i>Databases, Relational and Otherwise</i> (Datamation 1976).	270
Abb. 72: REDAMS (Keller/Starruß/Belke 1978, 3).	271

Abb. 73: Einordnung von REDAMS. (Keller/Starruß 1980, 17).	273
Abb. 74: Vera Wenks Matrix von Veröffentlichungen zur relationalen Logik (Wenk 1981, 10f.).	275
Abb. 75: Ausleihzettel der Dissertation Wenk, Universitätsbibliothek Dresden (Wenk 1981).	275
Abb. 76: ESER EC-1022 Großcomputer (Центр государственного хранения космической документации 1982).	276
Abb. 77: Dateneingabe am ESER EC-1022 Großcomputer, 1982 (Центр государственного хранения космической документации 1982).	276
Abb. 78: Verschiedene Optimierungsstrategien(Starke (Demuth) 1983, 43).	279
Abb. 79: Schichtenmodell einer relationalen Datenlogik (Starke (Demuth) 1983, 70).	280
Abb. 80: K 1600 als Beispiel für das System der Kleinrechner (SKR) der RGW-Staaten (Giebler 1980, 39–40).	283
Abb. 81: Funktionsschema von DABA 1600 (Schubert 1985, 111).	286
Abb. 82: Performance-Benchmark (Menhart/Stumpf 1987c, 6)	287
Abb. 83: Automatisierung koordinatorischer Praxen (Autor).	294
Abb. 84: Instabile Stabilität – reversible Historizität (Autor).	297
Abb. 85: Von der spezifischen Datenbankanwendung zur allgemeinen Datenbanksoftware (Autor)	298
Abb. 86: Parametrisierbare Softwarefigurationen in SOPS und SAP (Autor).	298
Abb. 87: Trias aus Informationsmodell–Daten–Algorithmen (Autor).	299
Abb. 88: Einschlüsse/Ausschlüsse von Realität im Informationsmodell (Autor).	299
Abb. 89: Rationalisierung durch Automatisierung der Informationsflüsse (Autor).	301
Abb. 90: Künstliche ›Intelligenz‹: Tabellarische Informationsstruktur für Tencent ML-Images (Tencent 2018).	307
Abb. 91: Austausch von Information zwischen ›Neuronalem‹ Netzwerk und Datenbankmanagementsystem. (Autor).	308

# Abkürzungsverzeichnis

ACM	Association for Computing Machinery – Berufsverband der Computerproduzenten (USA).
AdW	Akademie der Wissenschaften (DDR).
AIDOS	Automatisiertes Informations- und Dokumentationssystem (DBMS, DDR).
ALLDBS	Oracle 5.1 Adaption (DBMS, DDR).
ARPA	Advanced Research Projects Agency – Forschungsleitstelle des US-Verteidigungsministerium, gegründet 1958, heute Defense Advanced Research Projects Agency (DARPA).
ASU	АСУ, Автоматизированная Система Управления – Typenorientierte Betriebssoftware (UdSSR).
BASTEI	Bankspeicherung Technischer Informationen (DBMS, DDR).
BOMP	Bill of Materials Processor – betriebliche Software zur Stücklistenverwaltung von IBM (USA).
C	Programmiersprache (USA).
CAD	Computer Aided Design.
CAM	Computer Aided Manufacturing
CICS	Customer Information Control System – Transaktionsmonitor von IBM ab 1972 (USA).
COCOM	Coordinating Committee on Multilateral Export Controls – Einigte sich auf Exportbeschränkungen von Hochtechnologie in sozialistische Länder vor dem Hintergrund des Kalten Krieges.
CODASYL	Conference on Data Systems Languages – Netzwerk zur Koordination der Weiterentwicklung von Programmiersprachen. Zu CODASYL gehörten auch Untergruppen, so die Data Base Task Group, welche sich der Weiterentwicklung von DBMS widmete.
CONCOMP	Research in Conversational Use of Computers – Forschungsprojekt an der University of Michigan (USA).
COPICS	Communication-Oriented Production Information Control System von IBM (USA).
CP/M	Control Program for Microcomputers – Betriebssystem (USA).
CRM	Customer Relationship Management – Betriebswirtschaftliche Software für Kundenbeziehungen.
DABA 1600	DatenBank 1600 für den Computer K 1600 (DBMS, DDR).
DAKS	Datenkommunikationssystem (Transaktionsmanager, DDR).
DB	Datenbank.
DB2	Relationales DBMS von IBM ab 1983 (USA).
DBBS	Datenbankbetriebssystem – Hardwarenahe Komponente (ohne User-Interface) eines DBMS.
DBMS	Datenbankmanagementsystem – jener Teil der Datenbanksoftware, welcher die grundsätzlichen Funktionen zur Verfügung stellt, im Unterschied zur Datenbankanwendung.
DBS/R	Datenbanksystem/Robotron (DBMS, DDR).
DOS	Disk Operating System – Betriebssystem für Personalcomputer (USA).
EDV	Elektronische Datenverarbeitung.
EDVA	Elektronische Datenverarbeitungsanlage.
ERP	Enterprise Resource Planning – Betriebswirtschaftliche Software zur betriebsinternen Ressourcenplanung (Arbeitskräfte, Materialien, Produktionsabläufe).
ESER	Einheitliches System Elektronischer Rechner (RGW).
ESY/R	Exekutivsystem/Robotron (Transaktionsmanager, DDR).
IBM	International Business Machines.
IBM 305 Ramac	Erster Festplattenspeicher 1956 von IBM (USA).
IBM/360	Großrechner 1964–1978.
ID	Identification – ID-Schlüssel sind durch Programme generierte Zeichenkombinationen, welche einen Datensatz eindeutig identifizieren.
IDS	Integrated Data Store – netzwerkorientiertes DBMS der Firma General Electrics (USA).
IMR	Institut für Maschinelle Rechentechnik, Dresden (DDR).

IMS	Information Management System – hierarchische Datenbank von IBM ab circa 1966, hervorgegangen aus BOMP (USA).
ING_DB	Ingres 5 Adaption (DBMS, DDR).
INGRES	Interactive Graphics Retrieval System – relationale Datenbank, entwickelt an der University of California, Berkeley (USA).
INTERBAS	In Entwicklung befindliches multimodales DBMS der späten 1980er Jahre (DBMS, DDR).
KME	Kombinat Mikroelektronik Erfurt (DDR).
MIMER	Relationales DBMS benannt nach dem nordischen Gott Mímisbrunnr (DBMS, Schweden/DDR).
MRP	Material Requirements Planning; Material Requirements Planning – Betriebswirtschaftliche Software der 1960er und 1970er Jahre, die in heutige ERP, CRM und SCM-Systeme übergegangen ist. Diente der Auflösung von Stücklisten in der Produktion.
MTS	Michigan Terminal System – Universitäres Forschungsnetzwerk (USA).
NÖS	Neues Ökonomisches System (DDR).
NSW	Nicht Sozialistisches Wirtschaftsgebiet (DDR).
OGAS	ОГАС – Общегосударственная автоматизированная система учёта и обработки информации, Gesamtstaatliches Automatisiertes System der Rechnungsführung und Informationsverarbeitung (UdSSR).
OLAP	On-Line Analytical Processing – Datenbankbasiertes Softwarewerkzeug zur Echtzeitanalyse von Daten.
OLTP	On-Line Transaction Processing.
Oracle	Relationales Datenbankmanagementsystem (USA)
PDP-11	Kleinrechner der Firma DEC 1970–1990.
PICS	Production Information Control System (USA).
RAND	Research and Development – Thinktank, gegründet 1948 (USA).
RGW	Rat für gegenseitige Wirtschaftshilfe der sozialistischen Länder.
RPD	Robotron-Projekt Dresden (DDR).
SABRE	Semi-Automatic Business Research Environment – Kommerzielles Flugbuchungssystem (USA).
SAP	Systeme, Anwendungen und Produkte in der Datenverarbeitung – modulare betriebswirtschaftliche Software der SAP AG (BRD) und gleichzeitig Firmenname.
SAWI	Speichern und Automatisches Wiederauffinden (DBMS, DDR).
SCM	Supply Chain Management – Betriebswirtschaftliche Software zur Material- und Logistikplanung.
SCP	Single User Control Program. Betriebssystem, Software welche die Grundkomponenten eines Computers miteinander koordiniert (DDR, CP/M kompatibel).
SDC	System Development Corporation – weltweit erste eigenständige Softwareentwicklungsfirma, ab circa 1955, zeitweise zu RAND Corp. gehörig (USA).
SED	Sozialistische Einheitspartei Deutschlands, die maßgebliche politische Formation in der DDR.
SKR	System Kleinrechner, Pendant zum ESER (DDR).
SOPS	Sachgebietsorientierten Programmiersysteme – modulare betriebswirtschaftliche Software des VEB Robotron (DDR).
SQL	Structured Query Language – Abfragesprache für relationale DBMS (USA).
SZS	Staatliche Zentralverwaltung für Statistik (DDR).
TGL	Technische Normen, Gütevorschriften und Lieferbedingungen DDR (Pendant zur DIN).
VEB	Volkseigener Betrieb (DDR).
ZFT	Zentrum für Forschung und Technik, Betriebsteil von Robotron, Dresden (DDR).
ZK der SED	Zentralkomitee der Sozialistischen Einheitspartei Deutschlands (DDR).
ZPD	Zentralstelle für Primärdokumentation (DDR).



## Nachweise

Eine gekürzte und überarbeitete Version des Abschnitt 2.3.6.2 *Priorisieren, koordinieren und parallel verarbeiten* erschien in:

Hunger, Francis: Transaktionsverarbeitung in relationalen Datenbanken – Zur Materialität von Daten aus Perspektive der Transaktion, in: Balke, Friedrich, Bernhard Siegert und Joseph Vogl (Hrsg.): *Kleine Formen. Archiv für Mediengeschichte*, Berlin: Vorwerk 8 Verlag, 2021, S. 101–111.

Eine stark gekürzte und überarbeitete Version des Abschnitt 3.3 *Kalifornien, Technische Universität Dresden und Robotron Dresden: Das Problemorientierte Seminar »Datenbanken« und DABA 1600* erschien in:

Hunger, Francis: Sozialistische Co-Innovation – Wie in der DDR die relationale Datenbank DABA 1600 entwickelt wurde, *Zeitschrift für Medienwissenschaft* 27 (2022), S. 65–78.

# Originalsprachliche Zitate

- 
- <sup>i</sup> Around 1964 a new term appeared in the computer literature to denote a new concept. The term was ›data base‹, and it was coined by workers in military information systems to denote collections of data shared by endusers of time-sharing computer systems (McGee 1981, 505)
- <sup>ii</sup> source of a fundamental cultural transformation (Castells 2010, xviii)
- <sup>iii</sup> network capitalism (Fuchs 2008, 104)
- <sup>iv</sup> There is a velocity, idiosyncrasy, and thickness to the changes wrought by software that makes it a fundamentally tricky phenomenon, potentially rich rather than inherently reductive, but not automatically so (Matthew Fuller, In: Chun 2011, viii).
- <sup>v</sup> the asserted, but not quite proven, superiority of computationally optimized forms of calculation (Fuller u. a. 2011)
- <sup>vi</sup> the tacit assumption that computational artefacts can do things smarter (Fuller u. a. 2011)
- <sup>vii</sup> both concepts are described by reference to materiality, interconnectedness, and processuality (Schabacher 2020, Fol 1, pre-print)
- <sup>viii</sup> index cards, writing tools, typewriters, discourse operators (including quotation marks), pedagogical media such as the blackboard, various unclassifiable media such as phonographs or stamps, instruments such as the piano, and disciplining techniques such as alphabetization (Siegert 2015, 2)
- <sup>ix</sup> we need to regard technical processes not as products of intelligence but as practices of skill (Ingold 2001, 29)
- <sup>x</sup> The origin of DBMS can be traced to the data definition developments, the report generator packages, and the command-and-control systems of the fifties—a time when computers were first being used for business data processing. (Fry/Sibley 1976, 19)
- <sup>xi</sup> Infrastructure typically exists in the background, it is invisible and it is frequently taken for granted (Bowker u. a. 2010, 98)
- <sup>xii</sup> If infrastructure makes worlds, then software coordinates them. (Rossiter 2016, xv)
- <sup>xiii</sup> the banality of spreadsheets, enterprise resource planning (ERP) systems, and software applications (Rossiter 2016, xvi)
- <sup>xiv</sup> The unobtrusive grayness of so many types of media practice, from system administration to data gathering or the control and verification of all sorts of qualities and attributes, calls for a kind of suspicious attentiveness, the cultivation of a sensibility able to detect minor shifts of nuance, hints of a contrast where flatness would otherwise be the rule. (Fuller/Goffey 2012, 12)
- <sup>xv</sup> Logistical software functions as a technology of governance and control, measuring the productivity of labor using real-time KPIs [Key Performance Indicators – F.H.]. Central to logistics is the production of new subjectivities of labor (Rossiter 2016, 9).
- <sup>xvi</sup> As in any area of history, from the vantage point of a different present we see new things in the past. (Haigh 2018, 4)
- <sup>xvii</sup> It is for media archaeologists as it was for Foucault: all archaeological excavations into the past are meant to elaborate our current situation. (Parikka 2012, 6)
- <sup>xviii</sup> We no longer find ourselves dealing with the mass/individual pair. Individuals have become ›dividuals‹ and masses, samples, data, markets, or ›banks‹ (Deleuze 1992, 5).
- <sup>xix</sup> The dramatic advance of computer and communication technology is used to justify claims that society is being transformed by technology at an unprecedented rate, even though technological change and productivity growth in recent decades are weaker now [in the West, F.H.] than at any time since the Industrial Revolution. (Haigh 2018, 11)
- <sup>xx</sup> A couple of us from the [IBM] Systems Department had tried to read it [Codd's 1970 paper] – couldn't make heads nor tails out of it. [laughter] At least back then, it seemed like a very badly written paper: some industrial motivation, and then right into the math. [laughter]. (Irv Traiger, in: McJones 1997, 8)
- <sup>xxi</sup> This paper is concerned with the application of elementary relation theory to systems [...]. (Codd 1970:377)
- <sup>xxii</sup> [The relational model] provides a means describing data with its natural structure only: that is, without superimposing any additional structure for machine representation purposes. Accordingly, it provides a basis for a high level [information] retrieval language which will yield maximal independence between programs on the one hand, and machine presentation and organization of data on the other. A further advantage of the relational view is that it forms a sound basis for treating derivability, redundancy, and consistency of relations [...]. (Codd 1969, 1)
- <sup>xxiii</sup> It is also the first book on the relational model by the originator of that model. All the ideas in the relational model described in this book are mine, except in cases where I explicitly credit someone else. (Edgar F. Codd 1990, V)
- <sup>xxiv</sup> File Management systems were designed around tape storage [...]. DBMS were designed around disk storage. (Haigh 2009, 13)
- <sup>xxv</sup> use of computers as assistants in the logical analysis of large collections of factual data (Levien and Maron 1967, 715).
- <sup>xxvi</sup> [...] the principal application of relations to data systems has been to deductive question-answering systems. Levien and Maron provide numerous references to work in this area. (Codd 1970, 377)

---

<sup>xxvii</sup> All data about a particular individual, for example, should be retrievable without recourse to a sequential file search. Thus, the data base should contain  $10^5$  or  $10^6$  elementary data items arranged in a single-level pool for random access on the basis of content. (Levien/Maron 1967:716)

<sup>xxviii</sup> The disk tracks are subdivided into sectors. There are ten sectors on a track; five on the top of the disk and five on the bottom. Each sector can store a 100-character accounting record. The record is stored as a series of magnetic spots recorded in the sector on the track. By storing ten 100-character records on each of 100 tracks on each of 50 disks, it is possible to store 5,000,000 alphanumeric characters in the unit. (IBM 1957, 12)

<sup>xxix</sup> Since we are concerned with the logical analysis of factual data, it is natural to look to formal logic for the structure of an information language. In the RAND Relational Data File we use that part of modern logic called the 'predicate calculus'. It permits description in terms of *properties* and *relationships* among *individuals* (Levien/Maron 1967:717).

<sup>xxx</sup> A common requirement is for elements of a relation to cross-reference other elements of the same relation or elements of a different relation. Keys provide a user-oriented means (but not the only means) of expressing such crossreferences (Codd 1970, 380).

<sup>xxxi</sup> Thus, we employ a relational information language [...]. (Codd 1970, 380)

<sup>xxxii</sup> Data entry directly in the relational information language would eliminate the need for subsequent translation; however, considerable effort would be necessary to identify, record, and check the large number of sentences needed to record the data; the probability of mistakes and omissions would be high; and changes in the information language would necessitate reextraction of the data from original sources. Data entry in the English language would be convenient for the input aides, but inconvenient for the computer. Consequently, we have chosen to use special input forms (Levien and Maron 1967, 718).

<sup>xxxiii</sup> For each author, *a*, whose name appears on a bibliographic form concerning publication, *p*, compose a sentence *a* / AU-THOR OF / *p* and enter it in the file (Levien/Maron 1967, 718)

<sup>xxxiv</sup> How can the language be made convenient for the nonspecialist user? (ibid. 721)

<sup>xxxv</sup> I am at loss to understand why there has not been more emphasis during the past six years by CODASYL on user other than the application programmer. (Codd et al 1975:132)

<sup>xxxvi</sup> It affords an immediate step, however, to associative indexing, the basic idea of which is a provision whereby any item may be caused at will to select immediately and automatically another. This is the essential feature of the memex. The process of tying two items together is the important thing. ¶ When the user is building a trail, he names it, inserts the name in his code book, and taps it out on his keyboard. Before him are the two items to be joined, projected onto adjacent viewing positions. At the bottom of each there are a number of blank code spaces, and a pointer is set to indicate one of these on each item. The user taps a single key, and the items are permanently joined. [...] ¶ Moreover, when numerous items have been thus joined together to form a trail, they can be reviewed in turn, rapidly or slowly, by deflecting a lever like that used for turning the pages of a book. It is exactly as though the physical items had been gathered together from widely separated sources and bound together to form a new book. It is more than this, for any item can be joined into numerous trails. (Bush 1945, pt. 7)

<sup>xxxvii</sup> Consider, for example, a data bank which contains information about parts, projects, and suppliers. The individual description for an individual object (such as a particular part) is called an entity. (Codd 1969, 2)

<sup>xxxviii</sup> Thus, we might say that data are fragments of a theory of the real world, and data processing juggles representations of these fragments of theory. (Mealy 1967, 525)

<sup>xxxix</sup> Frege's example of 'Evening Star' and 'Morning Star', and Russell's of 'Scott' and 'the author of Wauedey', illustrate that terms can name the same thing but differ in meaning. The distinction between meaning and naming is no less important at the level of abstract terms. The terms '9' and 'the number of the planets' name one and the same abstract entity but presumably must be regarded as unlike in meaning; for astronomical observation was needed, and not mere reflection on meanings, to determine the sameness of the entity in question. (Quine 1953, 21)

<sup>xl</sup> [...] in general, relations are correspondences between *n*-tuples from a set and *m*-tuples from some possibly different set. (Mealy 1967, 526)

<sup>xli</sup> In the first part of the paper, we have proposed a theoretical model for data and data processing. The model is a *system* of sets of *entities*, *values*, *data maps*, and *procedure maps*. The entities correspond to the objects in the real world about which data are recorded or computed. The data maps assign values to attributes of the entities; these maps are regarded as being sets of ordered pairs of entities and values, or *data items*. (Mealy 1967, 533)

<sup>xlii</sup> a clean separation of structural from other considerations (Mealy 1967, 529)

<sup>xliii</sup> 'Semantics' would be a good name for the theory of meaning, were it not for the fact that some of the best work in so-called semantics, notably Tarski's, belongs to the theory of reference. (Quine 1953, 130)

<sup>xliv</sup> Tarski's construction of truth is easily extended to other concepts of the theory of reference. It is a striking fact that these notions, despite the paradoxes which we associate with them, are so very much less foggy and mysterious than the notions belonging to the theory of meaning. (Quine 1953, 138)

- 
- <sup>xlv</sup> different procedures will, in general, want to access the same data in different ways and orders. The order in which data items are fetched and stored is (or should be) independent of the data organization [...] (Mealy 1967, 530)
- <sup>xlvi</sup> Dr. Westervelt had often lectured on the need for a general data structure that was not machine oriented but one that was information oriented, where design emphasis was placed on information content instead of machine representation. (Childs 1970, D6)
- <sup>xlvii</sup> The problem then is to devise or find a suitable structure for the information environment, such that the resulting information environment operations will insure the same result, independent of any particular machine-representation of such information. (Childs 1970, D-7).
- <sup>xlviii</sup> The concept of ›record‹ (as a sequence of bits or bytes) was well established at the time for representing ›data‹. The closest mathematical representation of a record was the 'n-tuple' in set theory. Thus set theory seemed like a good place to start an examination for a mathematical foundation for ›data structures‹. (Childs 2016b, per E-Mail)
- <sup>xlix</sup> A is a subset of B; A is equal to B; A and B are disjoint; A is an element of B. (Childs 1968, 5)
- <sup>i</sup> An STDS is essentially a meta-structure, allowing a question to 'dictate' the structure or data-flow. A question establishes which sets are to be accessed and which operations are to be performed within and between these sets. (Childs 1968, Teil Abstract)
- <sup>ii</sup> ›Searching‹ for data using mathematical concepts was quite acceptable, but ›modeling‹ data as a mathematical object was ›heresy‹. (Childs 2016c, per E-Mail)
- <sup>iii</sup> [...] as more users obtain concurrent access to a large data bank, responsibility for providing efficient response and throughput shifts from the individual user to the data system. (Codd 1970, 381f.)
- <sup>iiii</sup> The CONCOMP papers were unclassified but not freely available. Although Codd cited one of Childs' papers, I wonder how many people were able to read Childs' research at that time. The CONCOMP papers carried a restriction that they were available only to a ›qualified requester‹. (North 2010)
- <sup>liv</sup> The database problem came to him through a friend at IBM. This man was not only a friend, but he was a neighbor in New York. That's how Ted [Codd] took a look at what was involved and decided to apply set theory. His name was Charles Davies. It is however interesting that after Ted did his work on relations and relational databases, Davies said, Ted had stolen his ideas. But all that Ted did, was to apply the mathematics. The whole thing involved mathematics, so Davies' claim was rather silly (Interview Sharon Codd 2016).
- <sup>lv</sup> Ted was at that point located at Poughkeepsie [1966/67] and pushing his flat-file concept, that everything looked like a flat file. And Charlie was working with him. We didn't have a lot of database systems there, but we did have file systems and they have been working on file systems. (K. A. Salmond 2016)
- <sup>lvi</sup> Those application programs which take advantage of the stored ordering of a file are likely to fail to operate correctly if for some reason it becomes necessary to replace that ordering by a different one. (Codd 1970:378)
- <sup>lvii</sup> It was C[harles]. T. Davies of IBM Poughkeepsie who convinced the author of the need for data independence in future information systems. (Codd 1970, 387)
- <sup>lviii</sup> [...] the problems treated here are those of data independence – the independence of application programs and terminal activities from growth in data types and changes in data representation – and certain kinds of data inconsistency [...]. (Codd 1970, 377)
- <sup>lix</sup> Allocation of space in core memory and disk storage, assignment of input-output units, and control of time-sharing should be based upon the needs of the programs being executed (and not upon some rigid subdivision of the machine). (Edgar F. Codd et al. 1962, 194)
- <sup>lx</sup> It does seem that Frank Westervelt was unique in initiating the concept of ›data independence‹ which is the essence of the ›relational concept‹. No one else was even considering the concept at that time. Even Codd acknowledged that he got the idea in 1967 from C. T. Davis who had worked with Frank, and I got the idea directly from Frank in 1965. (Email with the author, Aug 10, 2016)
- <sup>lxi</sup> In relational algebra, the basic objects were tables, and you combined these tables with operations like joins and projections and things like that. The relational calculus was a kind of a strange mathematical notation with a lot of quantifiers in it. (Don Chamberlin In: McJones 1997, 11)
- <sup>lxii</sup> The relational model consists of a collection of relations, each of which can be thought of as a simple table. (Michaels/Mittman/Carlson 1976, 128)
- <sup>lxiii</sup> The most prominent feature of the period under consideration here is the tendency, culminating in *Principia mathematica* [by Russell/Whitehead], towards the logical systematisation of mathematics, and replacement of so-called ›intuitive‹ explanations and arguments pertaining to the elements of mathematics by formal proofs based on logically precise definitions or systems of axioms. (Bunn in: Grattan-Guinness/Bos (Hrsg.) 2000, 220)

---

<sup>lxiv</sup> Thus, the definition of a set can be understood as formally identical to the definition of a concept, or a general term. According to this attitude, we behold, in the structure of sets and of the otherwise undefined relation  $\in$ , the actual underlying formal structure of the relationship that holds between an object and a predicate or property that is asserted to hold of it. This is nothing other than the relationship of the universal to the particular, [...] (Livingston 2011, 44)

<sup>lxv</sup> [...] the possibility of eliminating nonsimple domains appears worth investigating. There is, in fact, a very simple elimination procedure, which we shall call normalization. (Codd 1970, 381)

<sup>lxvi</sup> Codd had a bunch of queries that were fairly complicated queries and since I'd been studying CO-DASYL, I could imagine how those queries would have been represented in CODASYL by programs that were five pages long that would navigate through this labyrinth of pointers and stuff. Codd would sort of write them down as one-liners. (Don Chamberlin In: McJones 1997, 10)

<sup>lxvii</sup> This interface representation consists of a rectangular array with the following properties: ¶ 1) data items are homogeneous within each column; ¶ 2) different columns may have dissimilar types of data items; ¶ 3) all rows consist of distinct tuples. (Codd 1971, 41)

<sup>lxviii</sup> A normalized relation can be viewed as a table of  $n$  columns and a varying number of rows (Boyce et al. 1973, 31)

<sup>lxix</sup> Ray and I were not mathematicians. We wanted to find a way to make access to relational data more accessible to people without formal mathematical training (Chamberlin 2016, per E-Mail)

<sup>lxx</sup> The problem is, many of them have not only been engineers, but have been in other disciplines, not in engineering or mathematics or anything scientific. And that was really the problem. I ran into English-Majors, people who have studied various history at college, and mathematicians weren't to be found. ¶ [...] And there he found people who understood what he was trying to do. Of course the reason for it was, that the people there were mathematicians. (Interview Sharon Codd 2016)

<sup>lxxi</sup> The ideas didn't have immediate appeal to practitioners, I think, largely because Ted couched them in mathematical symbolism and terminology. In his original query languages he used mathematical notation, like universal quantifiers and existential quantifiers, and he used Greek letters a lot. Things like that just give the appearance of something being very esoteric and difficult to deal with. Whereas, actually, what he was trying to do was to make queries easier to write, not harder. So I think the development of a language based on English keywords, which you could type on your keyboard and which you could read and understand intuitively, was a breakthrough that made it much easier for people to understand the underlying simplicity of Ted's idea. It didn't really make the ideas any more simple; it just made them look simple. (Chamberlin 2001, 9)

<sup>lxxii</sup> [...] a set of sets [...] (Mackenzie 2010, 8)

<sup>lxxiii</sup> Particularly appealing to us in this connection was the absence of a defined ordering to either the attributes or the tuples of a relation. (Darwen 2015, 2)

<sup>lxxiv</sup> Since relations are sets, all of the usual set operations are applicable to them. (Codd 1970, 383)

<sup>lxxv</sup> Such expressions are constructed from simple names of relations, relational operators such as =, logical connectives and the quantifiers of the predicate calculus (Codd 1969, 10)

<sup>lxxvi</sup> We realized that everything expressible using logical connectives had an equivalent expression using difference and/or intersection and/or union, but we didn't think that would be acceptable to our users (Darwen 2015, 3).

<sup>lxxvii</sup> [...] plurality of joins (Codd 1969, 7).

<sup>lxxviii</sup> It is a matrix of vertical columns and horizontal rows. (Goody 1977)

<sup>lxxix</sup> He has placed already at his right hand, in the lower space, a heap of pennies from 11 down; in the second, of shillings from 19 down; but in the third, of pounds. And this heap, indeed, ought to be opposite and directly facing him, for in the ordinary accounts of the sheriff it is the medium; in the fourth is the heap of the twenties; in the fifth, of the hundreds; in the sixth, of the thousands; in the seventh, but rarely, of the ten thousands of pounds; rarely, I say, that is when, through the king or his mandate, the account for the receipts of the whole kingdom is received by the nobles of the kingdom from the treasurer and the chamberlains. (Nigel 1180)

<sup>lxxx</sup> The genesis and genius of quantifying in Early Modernity is intimately tied to visualization. Visualization itself acquired the character of being a kind of language functioning at the two levels of representing *and* at the same time manipulating what was represented. (Krämer 2011:56)

<sup>lxxxi</sup> There are no tables to be found in the earliest written records of Sumer. The 5600 documents from late fourth-millennium Uruk and its neighbours ›serv[ed] the accounting needs of a complex administration including offices of the fisheries, of herded animals and animal products, of field management, grain production and processing, and of labor‹. They were formatted not into lines of ordered text but into boxes or ›cases‹, each of which represented one sense unit or accounting unit. [...] As literacy spread into other social domains—for recording legal decisions, royal deeds, and even poetry—it became increasingly imperative to order the written symbols to follow the structure of Sumerian as a spoken language. The visual logic of the earliest accounts was thus irrevocably lost in the shift to linear organization of writing during the first half of the third millennium BCE. (Robson 2003:21)

- <sup>lxxxii</sup> It [...] must show all places where work can be performed, and also the job in process, and at least one job ahead of each machine or workplace. (Alford/Bangs (Hrsg.) 1944, 188)
- <sup>lxxxiii</sup> Idleness is shown by color as follows: Labor, red—Material, black—Absence, yellow—Tools, blue—Machine, green. (Alford/Bangs (Hrsg.) 1944, 188)
- <sup>lxxxiv</sup> If the coordinate chequer were 200 km square in plan, there would be 3200 columns on the complete map of the globe. In the tropics the weather is often foreknown, so that we may say 2000 active columns. So that  $32 \times 2000 = 64,000$  computers would be needed to race the weather for the whole globe. (Richardson 1922, 122)
- <sup>lxxxv</sup> [...] this subdivision of employment in Philosophy, as well as in every other business, improves dexterity and saves time. Each individual becomes more expert in his own peculiar branch, more work is done upon the whole, and the quantity of science is considerably increased by it. (Smith 1776:12f.)
- <sup>lxxxvi</sup> I came across the chapter where the author treats of the division of work [travail]; citing, as an example of the great advantages of this method, the manufacture of pins. I conceived all of a sudden the idea of applying the same method to the immense work with which I had been burdened, and to manufacture logarithms as one manufactures pins. (de Prony 1824:4 zit. nach Grattan-Guinness 2003:108)
- <sup>lxxxvii</sup> The bureau marked a transition from small, informal computing teams to a large-scale highly organized bureaucracy. [...] de Prony consciously adopted an industrial metaphor. [...] The Bureau de Cadastre was in every sense a computing 'factory'. (Campbell-Kelly et. al. 2003:10f.)
- <sup>lxxxviii</sup> [...] by laying out the columns of the chosen values and the first row of entries, and preparing the instructions on the preparation of the remaining entries on the page. (Grattan-Guinness 2003:109).
- <sup>lxxxix</sup> The Mightiest Computing Team the world has ever known. (Grier 2003:284)
- <sup>xc</sup> As you will see, this story shows that physicists and electrical engineers had little to do with the invention of the digital computer – that the real inventor was the economist Adam Smith, whose idea was translated into hardware through successive stages of development by two mathematicians, Prony and Babbage. (Simon/Newell 1958, 2)
- <sup>xci</sup> The value of a table does not depend upon the difficulty of calculating it; for, once made, it is made for ever, and as far as the user is concerned, the amount of labour devoted to its original construction is immaterial. (Glaisher 1911:325).
- <sup>xcii</sup> When Sharon Weinberg, Ted Codd, and I were forming our consulting company, the question of a company logo came up, for use on business cards and the like. I said I thought the logo should incorporate a stylized picture of a table. Sharon agreed. Ted said: 'I don't get it—we're not in the furniture business, are we?'. (Date 2006, 10)
- <sup>xciii</sup> The goal was to give the user a conceptual model which was unsurprising – it was called the principle of least surprise. [...] Our [Visicalcs'] model was the spreadsheet – a simple paper grid that would be laid out on a table. (Frankston 2015)
- <sup>xciv</sup> Individuals have become 'dividuals' [...] (Deleuze 1992, 5)
- <sup>xcv</sup> [...] and masses, [have become] samples, data, markets, or 'banks'. (Deleuze 1992, 5)
- <sup>xcvi</sup> Ray and I noticed that the example relations in Codd's 1970 paper looked very much like tables. We thought that the best way to explain relational data was to use informal words like tables, rows, and columns, rather than mathematical terms like relations, domains, and tuples. We developed the SQUARE language around this 'table' concept. (Chamberlin 2016).
- <sup>xcvii</sup> And so for them he [Codd] said, what is the most elementary and intuitively understandable way of representing information? Surely, it is just tables. We use tables all the time, for all kinds of things. And I had observed myself, talking to people about their database requirements, and asking, 'What's your data look like?' They will start drawing tables, even if they're IMS users. (Haigh/Date 2007:23)
- <sup>xcviii</sup> [...] that would have resembled modern diary entries more closely than account ledgers. (Poovey 1998, 34)
- <sup>xcix</sup> Because the double-entry system assigned a proper place to every entry, it saturated place with meaning and rendered place a mechanism of detection; but because several authorized persons were allowed to write in the ledger, place did not reflect the status of the writer but made writers interchangeable, so that place became a substitute for status, not its reflection. (Poovey 1998, 35)
- <sup>c</sup> "Every transaction can also be judged according to its effect on the rate of return on capital (profit divided by opening capital). [...] for merchant capitalists the rate of return on capital is fundamental, not merely profits" (Bryer 1993, 114f.).
- <sup>ci</sup> Every transaction can also be judged according to its effect on the rate of return on capital (profit divided by opening capital) [...] For merchant capitalists the rate of return on capital is fundamental, not merely profits. [...] Thus, from Marx's perspective the following hypothesis is suggested: double-entry bookkeeping emerged as capital became socialized in response to a collective demand from investors for the frequent calculation of the rate of return on capital as the basis for sharing profits. (Bryer 1993:114f)
- <sup>cii</sup> »uniquely-specified algorithm« (ebd.)
- <sup>ciii</sup> To record all inventory was a massive undertaking: houses, lands, stores, commercial inventory, cash, receipts, and promissory notes. The inventory for one large house alone could entail numerous record books. To handle this information, the

---

merchant needed to master the discursive and indexing tools necessary for writing, summarizing, and making accounts accessible for easy reference. (Soll 2009:56)

<sup>civ</sup> As English monarchs were gradually convinced that numbers could be useful, both because they afforded precise records of money and therefore indexes to national wealth and because they could be treated as accurate records and therefore excuses for policy, they slowly became willing to support the individuals who made the numbers make sense. (Poovey 1998, 97)

<sup>cv</sup> You know, if you think about banking it really hasn't changed that much. Automatic clearinghouses work more or less the way they used to. The equipment is more modern and cheaper, but I suspect many of programs were written twenty or thirty years ago. (Gray 2002, 28)

<sup>cvi</sup> I was mortified once when I went into our Premium Collection section and saying to the girl, ›Well, what's your job here?‹, and she saying, ›I do CP11s‹, and I said, ›Well, what's a CP11?‹ and she said ›Well it's a transaction. I put this number in this box here on this piece of paper.‹ And I said, ›Well why are you doing that?‹ She didn't know. Not a clue. Because she had no need to know. The job had been reduced to the point where she didn't really need to know. Mind you, I don't think she enjoyed the job any more through not knowing. And I don't think she did it any better through not knowing. (Fincham 1994:205)

<sup>cvii</sup> Manual systems in use at the time maintained seat inventories for a given flight at its point of departure. Reservations agents were free to book space on a flight after confirming seat availability posted on large display boards in each reservations office. After selling a seat, an agent sent a one-way booking message via telephone or teletype to the reservations office in the flight's originating city. Upon receipt of the booking message, a clerk decreased the count of available seats for the flight from an inventory maintained in a loose leaf folder. When the number of available seats for a flight dropped below a specified level, a ›stop sale‹ message was sent to all reservations offices for posting on the availability boards. (Copeland/Duncan G./McKenney 1988, 354)

<sup>cviii</sup> These accurate passenger inventories afforded the opportunity to control under/overbookings to jointly optimize passenger service levels and load factors (the percentage utilization of available seat capacity). Passenger databases could be used to model alternative decision scenarios. (Copeland/Duncan G./McKenney 1988, 357)

<sup>cix</sup> The rate of fare changes shifted from semiannually to monthly, semimonthly, and in some cases daily. (Copeland/Duncan G./McKenney 1988, 362)

<sup>cx</sup> American also came to perfect its management of a complex fare structure and benefited considerably from using SABRE's downstream intelligence. (Copeland/Duncan G./McKenney 1988, 368)

<sup>cx<sup>i</sup></sup> The application programs themselves were to be ›re-entrant,‹ meaning that each routine would have to clean up its processing work even though the passenger booking was not yet complete. Under this multiprogramming approach, several reservation requests could be processed simultaneously, sharing such common tasks as checking availability, entering passenger data, or ordering special meals. Individual PNR data was maintained in an ›agent assembly area,‹ and only upon activation of the end transaction program would all the data for a particular PNR be filed away permanently. (Head 2002, 34)

<sup>cx<sup>ii</sup></sup> The words ›transaction‹ and ›task‹ have the same connotation in CICS and are used interchangeably throughout this publication. (IBM 1972)

<sup>cx<sup>iii</sup></sup> Because the clerk has direct access to all of these accounts, he can complete the posting of each transaction before beginning the posting of the next. This accounting method is called in-line processing. In-line processing has previously not been practical in automatic accounting systems because of the difficulty of reaching and changing single records in large files. (IBM 1957, 5)

<sup>cx<sup>iv</sup></sup> The IBM RAMAC is built around a random-access memory device that permits the storage of five million characters of business facts in the machine. (IBM 1957, 5)

<sup>cx<sup>v</sup></sup> Calculated addressing proved not to be practical on the RAMAC. Consequently, the RAMAC was not greatly successful.

<sup>cx<sup>vi</sup></sup> The Integrated Data Store, the first database management system, was a derivative, or a byproduct, of MIACS. (Bachman 2004, 45)

<sup>cx<sup>vii</sup></sup> These teletypes were scattered around the United States, at all of the Weyerhaeuser sales offices, mills, and warehouses. In a sense, this version of the Problem Controller was a real message-oriented operating system. Messages could come in and be sent out to somebody else, or they could come in and be transferred to the GE 235 computer to be dispatched by Problem Controller. (Bachman 2004, 83)

<sup>cx<sup>viii</sup></sup> If you think about being a list or a chain of things, you'd like to have a purchase order that is associated with its line item records. I want to go from the purchase order to the first line item record, to the second line item record, . . . to the last line item record have it points back to the purchase order. I can start with any line item and get to the purchase order. (Bachman 2004, 53)

<sup>cx<sup>ix</sup></sup> Two orders from two different sales offices, two orders from the same sales office, it didn't make any difference. They're distinct transactions, and you want each one to run as if it was in a database all by itself. The result you want is to give each program database access, as if there were no interference, from any other program. The goal was that the application

---

programmer and the programs that they wrote could assume that they existed all alone in the database without interference from any other program. (Bachman 2004, 88)

<sup>cxx</sup> Really, it was transactions that were waiting, not programs. Programs are only the processing rules, the transactions are really what are active or waiting. (Bachman 2004, 90)

<sup>cxxi</sup> Cards were used to coordinate supplies from varied sources at varied times and places for timely assembly. Move cards directed output from a process onward to downstream activities. The cards provided management feedback by being passed back and forth between the Planning Office and the shop floor as the work was first issued, as it progressed through the required processes and the completion of each reported. (Wilson/James M. 2016)

<sup>cxxii</sup> A sphere of control has associated operations: activate SOC, back out SOC, recall SOC, terminate SOC, acquire resource (intent of use) and release resource (actual use). (Davies 1973, 140)

<sup>cxxiii</sup> An inquiry, a transaction, a job-step, a job and the day's work, can all be nested spheres of control if it is desirable, e.g., rerun the day to a different conclusion, at the expense of forcing no commitments until the end of the day. (Davies 1973, 141)

<sup>cxxiv</sup> To a large extent, the acceptance and value of the relational approach hinges on the demonstration that a system can be built which can be used in a real environment to solve real problems and has performance at least comparable to today's existing systems. (Astrahan u. a. 1976, 97)

<sup>cxxv</sup> The system state is not static. It is continually undergoing changes due to actions performed by processes on the entities. Read and write are examples of such actions. (Eswaran u. a. 1976, 624)

<sup>cxxvi</sup> The system state is not static. It is continually undergoing changes due to *actions* performed by processes on the entities. Read and write are examples of such actions. We assume that actions are atomic; that is, if two processes concurrently perform actions, the effect will be as though one of the actions were performed before the other. (Eswaran u. a. 1976, 624)

<sup>cxxvii</sup> This scheme involves a hierarchy of locks, with several different sizes of lockable units, ranging from individual records to several tables. (Chamberlin et. al. 1981, 639)

<sup>cxxviii</sup> [...] the actions of a process are grouped into sequences called *transactions* which are units of consistency. (Eswaran u. a. 1976, 624)

<sup>cxxix</sup> [...] consistency requires that a transaction be divided into a growing and a shrinking phase. During the growing phase the transaction is allowed to request locks. The beginning of the shrinking phase is signaled by the first unlock action. (Eswaran u. a. 1976, 627)

<sup>cxxx</sup> Unlike many companies, IBM maintains a Research Division with laboratories that are distinct from its product development labs. In the days of System R, IBM Research consisted of three laboratories, located in Yorktown Heights, San Jose, and near Zurich. ... All of IBM's research laboratories have a dual role: to contribute to the advancement of science and to develop technology for future IBM products. Like university professors, research staff members at IBM are expected to attend technical conferences and are evaluated partly on their record of publications in peer-reviewed open literature. (Wade/Chamberlin 2012, 43)

<sup>cxxxi</sup> The guys from SDL (later Oracle Inc.) were not academics, not mathematical theorists. They could not have developed SQL on their own. But once IBM did so, they knew what to do. Building a DBMS now would be like constructing a model airplane from a kit. (Wilson in: McJones 1997, 64)

<sup>cxxxii</sup> By 1978, it was clear that a relational system had significant advantages over the competing DBMS technologies, so it was expected the technology would be transferred to products. (Rowe 2012, 70)

<sup>cxxxiii</sup> A transaction is defined as the sequence of inserts, modifies, or deletes that have occurred to any file(s) in single node or multiple nodes in a network between begin-transaction and end-transaction system calls that transforms a data base from one consistent state to the next. (Schuster 1981, 11)

<sup>cxxxiv</sup> the major paradigm for synchronization and recovery in advanced database systems. (Haerder/Reuter 1983, 288)

<sup>cxxxv</sup> A growing number of executives are thinking in these terms: »We have our operational data under control, the core of our business where we keep track of our orders. Now we want to set up a system to gain insight into the strategic value of the data. Who are our customers and how should we segment our markets? Which of our customers return the most profit? What are product purchasing trends? Can we use our knowledge of our customers' needs to advise them, supply more of our products and simultaneously improve our perceived value to them, before our competitors beat us to it?«. (O'Neil In: Gray (Hrsg.) 1991, 1)

<sup>cxxxvi</sup> »Marketing Information Systems, Decision Support Systems, Management Reporting, or Direct Marketing« (ebd.)

<sup>cxxxvii</sup> What was the total revenue for the eastern region in the third quarter? (Forsman 1997)

<sup>cxxxviii</sup> What would be the effect on soft drink costs to distributors if syrup prices went up by \$.10/gallon and transportation costs went down by \$.05/mile? (ibid)

<sup>cxxxix</sup> Instead of columns and rows, users would get a puzzle of information, in which every multidimensional block is connected and associated to every other block. While a column might contain information on product names, sales revenue and



---

expenses for a row of products (such as nails, screws, hammers), OLAP tools would add dimensions such as time periods, sales territories and distribution channels. (Codd/Codd/Salley 1993, 88)

<sup>cxl</sup> the »dynamic principles« of systems thinking identified by Senge, »that can actually help us to deal with complex business environments.« (ibid., 3).

<sup>cxli</sup> Every transaction generates information about itself that is then used to guide further consumer choices (Bauman/Lyon 2013, 104).

<sup>cxlii</sup> Software may also be embargoed in its own right [...]. Examples include certain types of computer aided design (CAD) software, software related to the security of government classified materials, certain types of sophisticated development, programming, diagnostic or maintenance, and operating systems. [...] Software that is in the public domain and standard commercially available software is not embargoed (Van Kerckhove 1991, 11)

<sup>cxliii</sup> The process of developing a set of specifications for a complex hardware system by an orderly examination of specimens of that system (Rekoff 1985, 244)

<sup>cxliiv</sup> As an exception to these limitations, users of the East German [Robotron] ES-1040s may be able to employ directly IBM-360 applications programs. East Germany seems to have duplicated the IBM operating System successfully. (Central Intelligence Agency 1977, 7)

<sup>cxliv</sup> The rise of the relational database was a major phenomenon of corporate software in the 1980s and the 1990s. [...] A third force shaping the corporate software industry was Enterprise Resource Planning software, the applications software phenomenon of the 1990s. (Campbell-Kelly 2003, 169)

<sup>cxlvi</sup> The Planning Office failed for many reasons but a significant factor was its heavy demand on manual information handling and processing. (Wilson/James M. 2016)

<sup>cxlvii</sup> Given the enormous influence and widespread industry use of SAP software, we need to ask why this company and its products are so unknown by scholars in the fields of media studies, digital humanities, software studies, and network cultures. (Rossiter 2016, 52)

<sup>cxlviii</sup> So have ERP systems finally solved the materials control problems in large manufacturing organizations? To the degree that manufacturers experienced these problems in the 1950s and 1960s, the answer is an unequivocal yes. Today, lead times of more than a year or even several months for professional products are unacceptable (Peeters 2009, 66).

<sup>cxlix</sup> Everyone thought that the engineering function was much too complicated to be standardized across market place product lines (Thomas 2008, Abschn 16).

<sup>cl</sup> The Copics books were written at the Manufacturing Industry Center at IBM Munich by a group of experts including PICS team members Thomas Glaza and Don Ralston with minor contributions from [Joseph] Orlicky and [Oliver] Wight, and published anonymously (Peeters 2009, 60).

<sup>cli</sup> Files constitute the foundation on which the superstructure of the application is built. [...] The effective operation and efficiency of an MRP system is, to a considerable degree, a function of *system file quality*. (Orlicky 1974, 189)

<sup>clii</sup> [...] fixing something is one thing, but keeping it fixed is another. (ibid. 190)

<sup>cliii</sup> These are conflicting objectives. (ibid. 192)

<sup>cliv</sup> *common* base (ibid. 203)

<sup>clv</sup> Equally as important as the technical aspects of preserving data integrity – or perhaps more so – ist the training, discipline, and attitude of people. [...] [They] must be educated to the fact that a computer's outputs cannot be better than its inputs. (ibid.)

<sup>clvi</sup> Before a computerised production control system was installed there, the air in the production control department was blue, with arguments commonplace. The system at the centre of these disagreements was a »typical 10 year-old system that was large batch oriented and which filled a lot of waste paper bins«. (Baxter 1984, 32)

<sup>clvii</sup> It is apparently common for a company to be using a particular system and to be unable to wrench itself out of a rut because the people who *know* what should be done cannot talk to the managers because of that company's politics (ibid.)

<sup>clviii</sup> The IBM's Copics publication in 1972 played a pivotal role as an open standard available to the general public. Starting from this publication, a kind of genealogical tree can be imagined that would show how knowledge of one system (and its shortcomings) gave rise to the market introduction of another, and so on (Peeters 2009, 65).

<sup>clix</sup> Economic mobilization during the Second World War brought an incredible increase in industrial output and placed a premium on the integrated planning of production and distribution. Work simplification plans, printed forms, organizational charts, process charts, and instruction manuals were produced for use on an unprecedented scale. This wartime experience impressed many administrators with what could be accomplished when organizational structures and procedures were carefully crafted to achieve specific ends, rather than accreting slowly over time. (Haigh 2001, 18)

<sup>clx</sup> When Dietmar [Hopp] asked the group specific questions about the sales statistics system, it became clear that they had no clue what they were up to or what they were building. (Plattner 2012, Fol 2)

---

<sup>clxi</sup> We believed that most business processes we knew about, especially the whole world of transactional processing, could be done in real time. [...] We chopped the batch processes up and let them run in real time—in quasi-real time, so that we reduced the amount of batch processing to a large extent. Later, we created new batch applications. So, we were not yet 100% successful. But all data entry was 100% real time, which led to a system in which once data has been captured, you can rely on the data. A multi-step process was no longer required to validate data and accept data. [...] This would make it possible for business processes to follow each other instantly (Plattner 2012, Fol 4).

<sup>clxii</sup> In any case, most of the CPU time required by SAP applications is used in the data base system and the transaction monitor (CICS, IMS). (SAP GmbH 1988, Fol 4)

<sup>clxiii</sup> With the macros (dynpro, PG, ABAP/3, DB-interfaces), SAP has always had its own programming environment. These functions, formerly available in separate formalism, are now available in unified syntax and enhanced useability within the ABAP/4 language (SAP GmbH 1988, Fol 3).

<sup>clxiv</sup> It arises from the tension between local, customized, intimate and flexible use on the one hand, and the need for standards and continuity (Star/Ruhleder 1996, 114).

<sup>clxv</sup> Employees will react positively both when high levels of formalization are associated with routine tasks and when low levels of formalization are associated with nonroutine tasks. If, however, work is too formalized for the task at hand – if there are too many procedures too rigidly applied – all the outcomes invoked by the critics of formalization should be expected. (Adler/Borys 1996, 65)

<sup>clxvi</sup> It is an unfortunate fact that a significant proportion of package implementation projects undertaken in the late 1990s massively overran time and cost budgets. It is also a fact that the reasons for overrun were often related to integrating the package with other application environments. This was not usually a technical issue but related to differences in semantics and business rules between different applications that were never intended to collaborate. (Spratt 2000, 66)

<sup>clxvii</sup> The daily problem of administering the thousands of goods movement transactions (i.e. issues, receipts, and orders) grew linearly with the numbers and quantities involved. They might therefore have been handled by increasing the staff and equipment involved, but the problem of planning and controlling production grows exponentially with the numbers involved. (Peeters 2009, 58)

<sup>clxviii</sup> Widespread faith in the practicality and desirability of a totally integrated management information system [...] was marked by faith in the virtues of bigness, central planning, rationality, and technology to solve all problems. (Haigh 2001, 57)

<sup>clxix</sup> [...] the computer was reinvented yet again. (Ceruzzi 2003, 110)

<sup>clxx</sup> Both in terms of infrastructure and epistemic cultures, the vector space abstracts and concretises spaces inside data. (Mackenzie 2017, 243)

<sup>clxxi</sup> [...] the profit generation for which it is destined. (Couldry/Mejias 2019, XIII)

# *Data Proxy*

## Documentation of an Artistic Data Dramatization Project

1 Conceptual Framework	1
1.1 Interdisciplinarity?	3
1.2 Artistic strategies	6
1.2.1 Media art, post media art and the consequences	6
1.2.2 Artistic continuities: overidentification and narration	12
1.3 »What that means to you« – Towards the dramatization of data practices	13
2 Vlogging on YouTube – How a platform shapes visual content	16
2.1 Vloggers' burnout and platform metrics	16
2.1.1 Algorhythmia	16
2.1.2 Agency and interpellation	18
2.1.3 Metrics, user interactions, and the production of value	19
2.2 Vlogging infrastructures	21
2.3 A survey of vlogging aesthetics	22
2.3.1 Talk in a static setup	28
2.3.1.1 Visual Language incl. Typography and Style	28
2.3.1.2 Performance	29
2.3.1.3 Cinematographic qualities, incl. composition, scenery, montage, and sound	31
2.3.2 Cinematography oriented in a dynamic setup	35
2.3.2.1 Visual Language incl. Typography and Style	35
2.3.2.2 Performance	36
2.3.2.3 Cinematographic qualities, including composition, scenery, montage, sound	37
2.3.3 Conclusion	39
3 Episodes	44
3.1 Episode Development Notes	44
3.1.1 Character and script development	44
3.1.2 Production	47
3.1.3 Editing and Postproduction	48
3.2 Episodes Outline	52
4 Summary	53
Bibliography	54
List of figures	59

# 1 Conceptual Framework

- Alexander Kluge: Earlier, you cited a sentence by Brecht: »Petroleum resists five acts.« How does he mean that?
- Heiner Müller: That's a problem.
- Alexander Kluge: So, he feels that petroleum presents a challenge to him?
- Heiner Müller: You can only do something dramatically with a process that makes sense in terms of subjects, individuals, biographies ... (Müller 1997).

*Data Proxy* is the artistic media archeology of a potential future, a speculative deep time of media, which is marked by an all-encompassing data production. Ongoing datafication is attracting increasing attention, as it is part of a deeper transformation of society. *Data Proxy* addresses changes in the public/private dichotomy (Poster 1995), the automation and intensification of labor (Rossiter 2016), and a trend towards data colonialism (Couldry and Mejias 2019). This artistic research project focuses on two crucial issues:

*Data Proxy* looks into the fringe field of thick data (Wang 2013), the off-label use of data creation (Duguay 2020), and data doubles (Lupton 2016), building on the notions of ›epistemic harvest‹ and ›computational capital‹ (Hunger 2018). To intervene, this research shifts the emphasis back to those who create and record data, machines and humans alike.

*Data Proxy* researches and appropriates elements of the habitus and visual language of successful vloggers, namely Anna's Analysis (veganism, van life, soul searching),<sup>1</sup> Marques Brownlee (tech),<sup>2</sup> Elle Mills (pranks, LGBTQ, coming-of-age),<sup>3</sup> Casey Neistat (travel, entrepreneurship),<sup>4</sup> Kelly Stamps (fashion, minimalism),<sup>5</sup> Sailing La Vagabonde (travel, family),<sup>6</sup> Tara Brabazon (academic life)<sup>7</sup> and Markiplier (gaming).<sup>8</sup> It decisively aims for an aesthetic of success, ignoring the low visual DIY quality of smaller, less successful (in terms of subscribers) vlogs.<sup>9</sup>

---

<sup>1</sup> Anna Reid, <https://www.youtube.com/c/VeganEarthSoul/>.

<sup>2</sup> Mark Brownlee, <https://www.youtube.com/c/mkbhd/>.

<sup>3</sup> Elle Mills, <https://www.youtube.com/c/ElleOfTheMills/>.

<sup>4</sup> Casey Neistat, <https://www.youtube.com/c/Caseyneistatofficial/>.

<sup>5</sup> Kelly Stamps, <https://www.youtube.com/c/KellyStamps/>.

<sup>6</sup> Riley Whitelum & Elayna Carausu, <https://www.youtube.com/c/SailingLaVagabonde/>.

<sup>7</sup> Tara Brabazon, <https://www.youtube.com/user/TaraBrabazon/>.

<sup>8</sup> Mark E. Fischbach, <https://www.youtube.com/c/markiplier/>.

<sup>9</sup> The selection is primarily US-centered, except for Sailing La Vagabonde and Tara Brabazon, who are based in Australia. While the selection is diverse, it continues to re-inscribe a Western narrative. A cross-cultural analysis remains a desideratum. Rieder et. al. suggest that YouTube follows a language hierarchy, with English at the top, followed by languages which are widely spoken (Rieder, Coromina, and Matamoros-Fernández 2020).

This artistic research project focuses on the aesthetic qualities and performative engagement of vlogging and the question of how to serve and appropriate this specific genre by artistic means. Vlogging is a »format that can embed multiple genres, such as recounts of daily life (e.g. procedural accounts of morning and bedtime routines), »updates« about events in the vlogger’s life, »confessional« videos saturated with emotion and »how to« videos showing how the vlogger undertakes, for instance, a domestic task. The latter differ from the general »tutorial« videos in their focus on the intimate and every day, and by embedding the instructions within videos of daily activities and routines (as opposed to a tutorial showing only how to do something outside its natural context)« (Zappavigna 2019, 3f.). Zappavigna’s definition should be expanded slightly, since vlogging can also include genres like gaming, travel and exploration, political comments (if they revolve around the authors’ subjectivity) or personalized live streaming. In contrast to Zappavigna, this paper also includes gaming, talk and technology-oriented formats, since these demonstrate a specific intimacy with games, technology or discourse. Vlogging is a subset of influencer culture, centered on video streaming platforms like Youtube, Twitch or TikTok. Influencer culture itself is much broader than vlogging alone and includes other social media platforms like Twitter, Facebook and Instagram with their respective content formats.<sup>10</sup>

While the first component of the Ph.D. thesis has involved academic empirical research through the lens of media theory, the second component documents an art project. It is clear that such documentation does not necessarily generate new knowledge. It offers a reflection on the methods and artistic strategy that emerged within the artwork itself. Thus, it is a shadow of the emergent knowledge that originates within an artwork. The documentation of the *Data Proxy* project emerges in three steps: First, it discusses the contextual artistic framework within which the work was developed. It gives an overview of the artistic strategy and shows the methodological framework. Second, it presents the background research: The project exists as a series of online videos on YouTube in a vlog format. The platform plays a major role in the project’s context. Therefore, it is important to emphasize that YouTube is the project’s platform because of the viewership and creator community, the comment section and the right margin recommendations, as well as the advertisement ecosystem. Third, this paper discusses the episodes of *Data Proxy* were developed, followed by the scripts for each episode. Readers of this document have to make one major decision. They can either read this documentation first or watch the project itself first, prior to reading the documentation. When it comes to watching, readers are invited to visit the following URLs first.

Episodes	YouTube
1 – More Data!!!	<a href="https://youtu.be/U0zq96xg6o0">https://youtu.be/U0zq96xg6o0</a>
2 – Data Toilet	<a href="https://youtu.be/mGbhEPcnaNo">https://youtu.be/mGbhEPcnaNo</a>
3 – Data Poor?	<a href="https://youtu.be/f3vLdrtPMao">https://youtu.be/f3vLdrtPMao</a>
4 – Meeting a Professor	<a href="https://youtu.be/lcJaMVD9f94">https://youtu.be/lcJaMVD9f94</a>
5 – 20K Eurocoin lost!	<a href="https://youtu.be/SD_N7IPcROM">https://youtu.be/SD_N7IPcROM</a>
6 – Data Socialist?	<a href="https://youtu.be/kTHV5ESfgTQ">https://youtu.be/kTHV5ESfgTQ</a>

---

<sup>10</sup> Compare: *The Like economy* (Gerlitz and Helmond 2013), *Status Update: Celebrity, Publicity, and Branding in the Social Media Age* (Marwick 2013), *Microcelebrity and the branded self* (Senft 2013), and *Wolves in Sheep’s Clothing? Influencer Marketing beyond Endorsements* (Rohde and Mau 2019).

## 1.1 Interdisciplinarity?

Since the relationship between academic, institutionally-sanctioned knowledge and artistic knowledge emerging from artistic practice has always been delicate, this section serves to address how my practice travels between the two.

The theoretical aspect established the foundations and created the material for my artistic practice. While the theoretical component is an exploration and application of scientific and knowledge-based methods of research, the artistic component provides a means for reflecting on previous results at an artistic and social level. It is aimed at making the socially implicit and tacit knowledge visible (Polanyi 1958) through artistic means.

This artistic research is specific insofar as it involves the intertwining of two different registers: the artistic practice, on the one hand, and an academic, empirically-based approach on the other. Contrary to the attempts of a large part of the artistic research community, my approach maintains a certain degree of separation between these two registers. Most broadly, this is due to an insistence on the production of different kinds of knowledge. While the former tends towards subjectified and embodied knowledge production, the latter is stronger in terms of the verifiable interconnectedness of discourse and in the exchange of knowledge. Granted that in current discourse, separation is always conceptualized as a continuum between what has been separated, I use these distinctions to achieve a sharpening of contexts and fields, making them identifiable as different discourses with different affordances.

Both components, artistic and scientific, are intentionally separated. This divided arrangement is intended to achieve clarity about the methods and means. It is designed to meet one of the most pressing concerns about an interdisciplinary<sup>11</sup> theoretical/artistic practice: »there can be a danger that artistic practice becomes an illustration of, or gets lost in, the processes of writing and engagement with theoretical concerns, rather than the necessary articulation of the doing of things that are inherent to artistic research. Often artistic researchers forget to name the most obvious (for example; what and how do I do what I do? what decisions have I made?), and become entangled with theorizing an area of interest« (Bacon and Middelow 2019, 39). In response to this, my research design has clearly separated both areas. Such a divided research design doesn't prevent interdisciplinarity, it rather sharpens the respective fields, creates separated and deeply-researched results and only allows for subsequent intersections. It prevents the ghosting of artistic practice by theoretical research as warned against by Bacon/Middelow.<sup>12</sup>

The research design, however, still allows for overlap: For example, the emphasis on the difference between data and non-data and how it gets separated through the information model in the theory component of this project was initially informed by practice-based explorations during the artistic media anthropological *Database Derive* workshop (Hunger 2015). Then again, media-theoretical academic research leaves a surplus, which cannot be expressed in academic writing. Epistemic surplus produces the art practices' potential. It also demonstrates where art can be more than an ›illustration‹ of theory,

---

<sup>11</sup> Arlander gives a definition of different concepts: »Interdisciplinary usually means integration of knowledge across academic disciplines, while multidisciplinary refers to the juxtaposition of different types of knowledges and transdisciplinarity denotes the production of knowledge with parties beyond the academy« (Arlander 2016, 3).

<sup>12</sup> For a critical reflection see *Interdisciplinarity – A Critical Assessment* (Jacobs and Frickel 2009, 47f.).

for it simply explores and researches capillaries, which cannot be accessed within the academic context of the social sciences today. Exploring and expressing this surplus creates research value in itself. By referring to Donna Haraway's feminist concept of »situated knowledge«, we can conclude one of the strengths of artistic research: Subjectified, embodied and performative artistic practice acknowledges contingency and situatedness, in order »to learn how to see faithfully from another's point of view« and to establish »the capacity to see from the peripheries and the depths« (Haraway 1988, 583).<sup>13</sup> Like Haraway, artist Barbara Bolt insists on a productive difference between both fields: while the academic, empirical approach describes and models the world, art research adds the performative creation of things within the world and creates ripple effects in the »material, affective and discursive domains« (Bolt 2008, fol. 13).<sup>14</sup> How these ripple effects shape research can be demonstrated with the example of *Data Proxy*. It is not just a video series pragmatically published on YouTube for broadcast. Rather, the deliberate decision to use what is currently the largest video platform places the work in the very same data ecology that it criticizes. The work itself becomes a testbed for the accumulation of views, the shaping of content and aesthetics to please the platform's demands, the importance of YouTube's recommendation algorithms, the invisible audience labor of the commenters and data creation through marketing-oriented data capture.

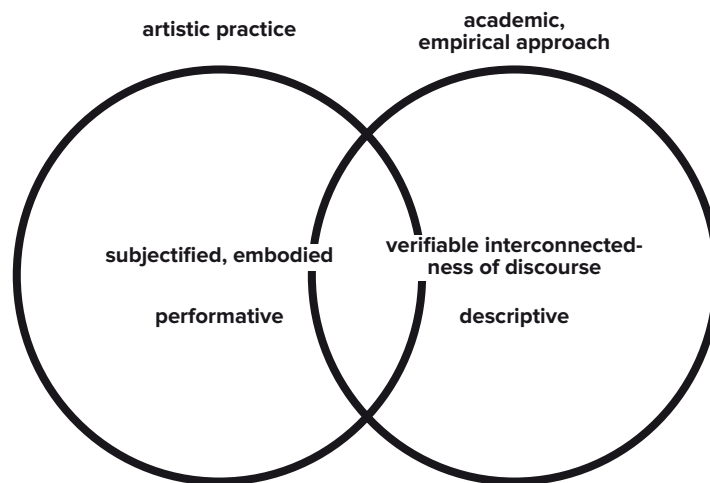


Fig. 1: The overlapping of artistic and academic empirical approaches. Separations of each knowledge domain remain and provide clarity about methodologies, while the overlapping of the two provides a space of disciplinary struggle and potentially new knowledge. The major difference lies in the degree of describing the world, since art not only aims to act in the world, but to intentionally intervene in it (author).

The next question is: To what extent does this shape an interdisciplinary approach? Is there a necessity, a need for an interdisciplinary approach? Does it generate knowledge and develop methodologies, which the individual disciplines are unable to create? Let's first look at how artistic practice benefits from media theory:

<sup>13</sup> A similar argument is developed in *What is media archaeology?* (Parikka 2012, 44–49).

<sup>14</sup> For a contextualization of Bolts' argument see *Artistic research and/as interdisciplinarity* (Arlander 2016, 12).

Over time, I have observed that many projects in the field of media, post-media, and post-internet art remain at the (visual) surface, delivering illustrated versions of what the op-eds and feuilleton momentarily discusses. For example, discourses around surveillance, the question of face recognition, or critical takes on social media platforms are currently prevalent. Critically examining the situation may lead to the question: How is it possible to research beyond these obvious discourses?

Media-theoretically informed media art can differ. Media scholar Erhard Schüttpehlz has offered a distinction between public media (or communication-oriented media) and infrastructural media, and pointed towards media of cooperation (Schüttpehlz 2006; 2016). To know or discuss this differentiation would serve the (post)-media art community well, since it makes it obvious how urgent a stronger artistic debate about cooperative media is. My own media theoretical research into databases demonstrates their substantial impact on everyday life, bringing to light the large extent to which infrastructural media enable contemporary society. In my case, this distinction led to a specific understanding of what data is: material that enables cooperation. This materialist turn of data is a departure from the doomsday ›data as a surveillance apparatus‹-critique that the majority of (post-)media art provides today. Media theory enables media art to create different works. It is distressing and irritating how little concern media art has for media theory.

To further expand on the theory-practice relation: I didn't read Couldry's/Mejia's book on data colonialism until I was nearing the end of the *Data Proxy* Project, long after the script had been developed and written. It is, however, exciting how my project overlaps with many theoretical elements of their work: The notion of ›data production‹ which runs through all episodes of the *Data Proxy* connects with what they call »data relations« (Couldry and Mejias 2019, 27–35), the »personal data store« as addressed in Episode 6 is mentioned and its ideology criticized (p.139), they mention »nudging« (ibid.) which appears in Episode 5, and the question of data rich/data poor is addressed by Couldry/Mejias, who note that the ›infrastructures of connection, like all forms of power, build on existing inequalities« (ibid., 24f.). In my reading, this overlapping demonstrates how art can capture the social subconscious, in this case by discovering, expressing and exposing existent social formations. The difference is that Couldry/Mejias develop their discussion as a critique, while the *Data Proxy* as a figure is far more indifferent and affirmative towards data production. It thus may be possible to intervene into social formations other than those of the more academic, post-Marxist reading of Couldry/Mejias.

One more momentous example for multi- or interdisciplinary work is the media-theoretically informed realization that an immanent digital aesthetics doesn't exist. All visualizations today are metaphors, mostly taken from earlier non-digital media, some from color theory. The computer itself is unaware of the meaning of what it processes. Meaning is always created by humans and for humans. Thus, every articulation of the ›digital‹ is a human design decision. Artistic works that apparently look very ›digital‹ may refer to the cultural effects of digitalization without grasping it. They rather use the unspoken agreement about metaphors between artists and spectators. Therefore, they do not actually address digitalization, but our phantasms around digitalization. In contrast to that approach, a media-theoretical realization of such differentiations shapes my artistic practice and calls for the cautious and intentional use of metaphors.

How, in turn, does the media theory side benefit from artistic explorations? The aforementioned example of the *Database Derive* (2015) workshop, with a media-anthropological methodology that makes it possible to understand to what extent databases pervade everyday life, illustrates the point. Another instance is the inclusion of a chapter on tabulation practices into the theoretical component. A focus on



tables would not have resulted without an awareness of the (visual) form, rooted in artistic and design practices. If the research into tables had only taken place in the artistic field, it would have concentrated on the visual and diagrammatic effects – that is how the chapter was initially written. Just when interdisciplinary influences emerged, I reformulated the subject towards a media practice of tabulating (the final form of the chapter). Still, the artistic project *Data Proxy* cannot claim to be art-practice-led research, since major parts of the theoretical body were already developed when I began working on it. These examples have shown how an overlapping approach was able to challenge the assumptions of each separate field and thus contribute to each field individually. It was also possible to demonstrate how the overlapping of both artistic practice and academic discourse creates new knowledge. Further it was argued that the artistic practice can potentially express ideas which appear in academic writing as well, simply using different means.

However, new methodologies through the ›inter‹ of disciplines have not emerged and thus it might be beneficial to frame the research process not as interdisciplinary and mark it instead as leaning towards the multidisciplinary: »Rather, the research value will come from studying the changes in perception which occur with the move [...] from one disciplinary perspective to the other« (Cazeaux 2008, 31). With Cazeaux, we can then argue that the intersecting lens shape in Fig. 1 is not so much new knowledge itself, but »occasions for negotiation between domains«, and a »tangled network of resistances and new possibilities which emerges from the negotiation« (Cazeaux 2008, 33). Does the *Data Proxy* project versus *The Form of the Database – Genealogies, Operationalities, and Genealogies in East and West* disappoint promises of interdisciplinarity in artistic research? If so, then the disappointment should be only partial. For both respective knowledge domains, media arts and media theory have been enriched in the process.

This section's discussion has already shown a certain discontent and uneasiness with what (post-)media art currently is able to contribute to our understanding of digitalization. It therefore makes sense to position my work within the field more precisely, which is the next section's task.

## 1.2 Artistic strategies

### 1.2.1 Media art, post media art and the consequences

It is now crucial to discuss the author's artistic practice in relation to concepts which have emerged over the past 20 years. These artistic concepts include differentiations among, for example, visual art (Bishop 2012), media art, new media art, and post media art (Apprich 2013; Broeckmann 2013; Quaranta 2013), net.art (Arns 2002, 66–71) or postinternet art (Olson 2011; McHugh 2011) and digital art (C. Paul 2011).<sup>15</sup>

---

<sup>15</sup> It is noteworthy that the notions of new media art, digital art and postinternet art have been developed against a North-American background and are partly centered around artists from the US. Media art and net.art have seen

My practice has been situated in Medienkunst or media art. According to Domenico Quaranta, Medienkunst comprises »all media: press, radio, fax, telephone, satellite communications, video and television, light, electricity, film, photography, *and* also computers, software, the web and video games« (Quaranta 2013, 23). Another definition is provided by Daniels/Frieling, who state that media art is »by definition multimedia, time-based or process-oriented« (Daniels, Frieling, and et. al. 2001). Their reading of Medienkunst is clearly rooted in modernity: »Tendencies of art and media technology development throughout the twentieth century serve as the background for promoting historic and contemporary perspectives on artistic work in and with the media« (ibid.). Medienkunst promotes the contemporariness of art through the usage of contemporary media. Yet its rootedness in modernity undermines using the term ›Medienkunst‹ in relational practices which have emerged since the 1990s. More broadly, my practice has involved many different media. Early on, I created net.art pieces (1999, 2004). Subsequently, I developed an interest in satellites (2001, 2007, 2010), used radio in two of my installations (2009, 2014), and created one radio play (2012). I addressed the very concept of digitalization in a completely analog performance, *The Setun Conspiracy* (2005). The installation and performance developed a narrative about a ternary (or nonbinary) Soviet computer of the 1950s, thus delving into the genealogies of the digital. These were explorations of the threshold between the analog and the digital: Eventually, analog media were converted to digital, as with radio to net.radio, analog satellite to digital and VHS and Betacam to DV tape. It looked into the genealogies of the computer meta-medium on the one hand, and into Eastern European modernity, on the other. This already rich exhibition practice was further diversified through a discursive stream: holding public lectures, curating exhibitions, and creating performances and workshops.

The guiding questions are: How does this broad range of media practices foster an interest in exploring different forms of expression? How have these forms of expression emerged? How do they overlap before and after the computer, the meta-medium? The aforementioned works offer a critical examination of the historiography of technology as ideologically-charged constellations of knowledge and power. The term ›media artist‹ might not completely fit with this approach since the visual element in my work was never limited to media technology. Many of my works thus integrate media into spatial exhibition practices. They react to a specific space and create their own space, e.g., through building dedicated projection screens, setting up a sculptural waiting line (Fig. 4) or using a large antenna as a spatial element (Fig. 5.).

---

a broader geographical spread, with the latter influenced by artistic practices from Eastern Europe over Central Europe to North America.

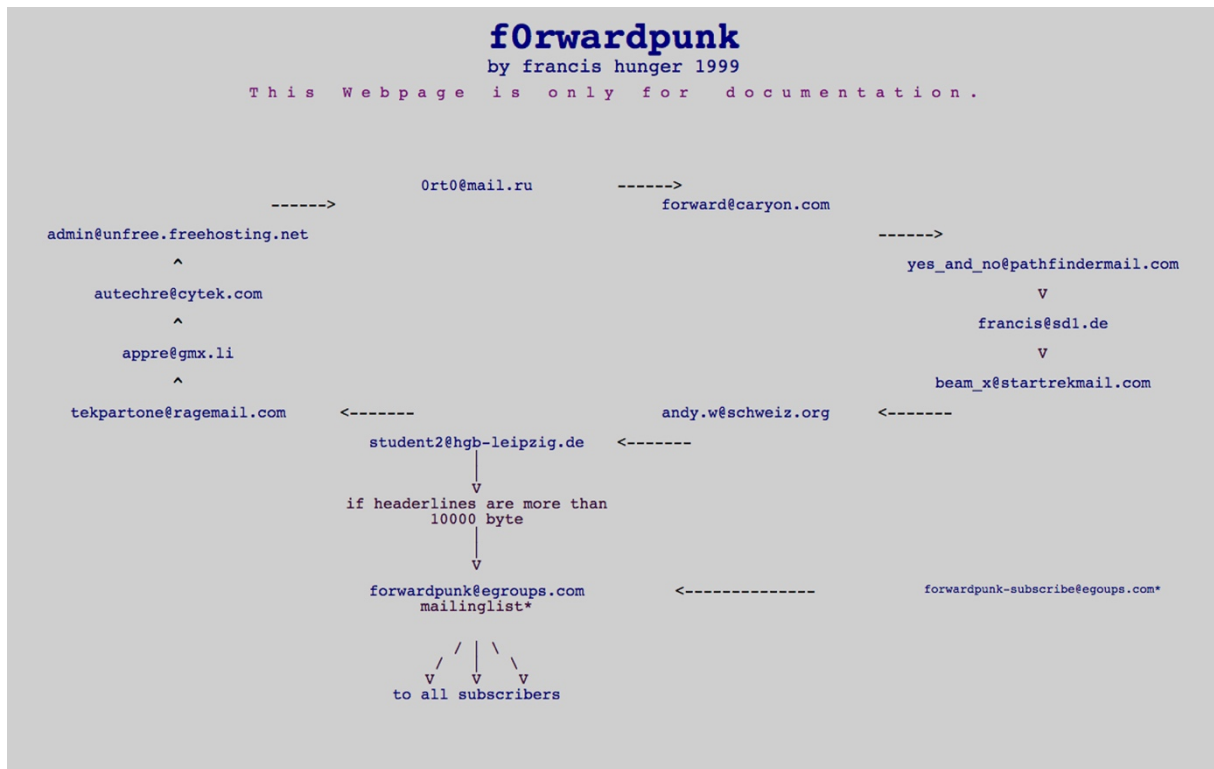


Fig. 2: *F0rwardpunk*. net.art project, 1999 (screenshot, author).



Fig. 3: *International Sputnik Day*. Installation and performance (digital prints, Fanzines, distributed online celebrations), installation view, Laden für Nichts, Leipzig, 2007 (Matthias Hennig).



Fig. 4: *The Setun Conspiracy*. Installation and performance (stanchions and walking line, cardboard sign, slides, cardboard office), installation view, Monitoring/Kunstverein Kassel, 2005 (Sven Heine).



Fig. 5: *History Exhaustion*. Installation (radio, antenna, sound, framed digital prints, book with narration, not depicted: three video performances), installation view, Galerie Metro, Berlin, 2009 (author).

Furthermore, the turn to narrative practices was not only grounded in the modernist concept of Medienkunst. Crucially, this turn was also motivated by the deficits in Medienkunst subfields, namely, generative, and interactive art. Artist Alexei Shulgin, arguing from the same Eastern European post-Stalinist background, warned against ›interactive art as manipulation‹: »Looking at very popular media art form such as ›interactive installation‹ I always wonder how people (viewers) are excited about this new way of manipulation on them. It seems that manipulation is the only form of communication they know and can appreciate. They are happily following very few options given to them by artists: press left or right button, jump or sit. Their manipulator artists feel that and are using seduces of newest technologies to involve people in their pseudo-interactive games obviously based on banal will for power.« (Shulgin 1997, 267). Shulgin's intervention characterizes the emergent media art of the late 1990s as a »transition from representation to manipulation« (ibid.).

How relevant these considerations are today is exemplified by the work *Future You* by the digital artist and designer collective Universal Everything, which was exhibited at The Barbican, London in the show *AI: More Than Human*, (May 16–Aug 26, 2019).<sup>16</sup> The work allowed humans to move in front of a screen that displayed an abstract structure, which would react to the interacting viewers, allegedly according to »Artificial Intelligence«. It is clear how problematic the claims around *Future You* are from the artists' introduction their work: »In *Future You*, you are faced with a unique reflection of your potential, synthetic self. Starting as a primitive form, it learns from your movements to adapt, suggesting an agile, superior version of you« (Universal Everything 2019). There are numerous problems with this statement, and this is not the place to discuss the social, artistic, and political consequences at length; instead the analysis will focus on the argument regarding the question of ›manipulation, not representation‹. In this piece, the promise of Artificial Intelligence boiled down to seven parameters for motion detection: size, style, color, material type, opacity, glow, and density (ibid.). These parameters translated into predetermined, parametric values to generate and adapt an abstracted limbic body (**Fehler! Verweisquelle konnte nicht gefunden werden.**). Based on these parameters, Universal Everything delivered an updated version of what Shulgin referred to as the »pure will for manipulation«, an updated »press left or right button, jump or sit« (Shulgin 1997, 267). *Future You's* claims, drawing on the buzz phrase of Artificial Intelligence, was that a machinic entity would create a »superior« version of the spectator/user. This message resembles the same narrow ideology of what humankind is, as communicated by Silicon Valley's industry proponents (c.f. Daub 2020). It is not only the artists who are to blame here, but also the institution and the curators who, sponsored by the Google Arts program, commissioned the work, and decided to provide the cultural industry with a spectacle version of what media art could be.

Now in the guise of AI, this tech demo scheme followed in the footsteps of previous generations of Internet, social media and Blockchain technology evangelists. Crucially, these claims did not deliver on the promises which made Medienkunst relevant in the first place: to decode technological figurations as social relations and provide insight into the rise of the digital society. Shulgin's intervention, emerging against the background of having grown up in a totalitarian state, has served as a consistent

---

<sup>16</sup> For imagery of the work see the website at <https://www.universaleverything.com/projects/future-you>. Art critique Jonathan Jones reviewed the show in *The Guardian I've seen more self-aware ants! AI: More Than Human – review* (Jones 2019).

warning to me. The question remains: What could be a way out? One response is the use of narration in my work, while keeping an eye on infrastructural issues. This approach resonates with what Claire Bishop in 2012 called the digital divide, with mainstream contemporary art on one side and new media on the other, creating two separate fields: »While many [mainstream] artists use digital technology, how many really confront the question of what it means to think, see, and filter affect through the digital?« (Bishop 2012).

Subsequent to pointing out basic decisions *ex negativo*, the following paragraphs discuss the stakes for my most recent work, *Data Proxy*. Since it makes use of a popular YouTube format, the vlog, *Data Proxy's* approach could be identified as ›postinternet‹, a term proposed by artist Marisa Olson in 2008. Postinternet refers to art after the Internet, and is not so much a technical as an aesthetic definition, where the selection or curation of available sources constitutes a major practice: »the work of pro-surfers transcends the art of found photography insofar as the act of finding is elevated to a performance in its own right, and the ways in which the images are appropriated distinguishes this practice from one of quotation by taking them out of circulation and reinscribing them with new meaning and authority« (Olson 2011, 60). And while *Data Proxy* picks up on a situation where the existence of the Internet is no longer new, and while it is interested in vernacular aesthetics, its vlog episodes are postinternet in form, not in content. How media ecologies came into being is irrelevant to postinternet art. Postinternet takes the medium simply for granted and works with its output.

Content-wise, *Data Proxy* and many of my previous works rather exhibit a dimension that is best called media-archeological. »Media archaeology sees media cultures as sedimented and layered, a fold of time and materiality where the past might be suddenly discovered anew, and the new technologies grow obsolete increasingly fast« (Parikka 2012, 3). Media archeology reformulates Foucault's archeology of culture and knowledge towards a media analytical view. An archeology of technology, as famously explicated by Kittler, serves to complement this methodology. *Data Proxy* simultaneously examines the materialities of a cultural practice and the materialities of technology (ibid. 2012, 163f.). To further develop this, we need Schüttpelz's differentiation between public media (mass media) and infrastructural media (Schüttpelz 2016). *Data Proxy* clearly uses mass media (YouTube and other social media) as the primary means of distribution. The issues, however, which *Data Proxy* researches and discusses lean towards the infrastructural dimension – that is, data.

Real world problems are of central interest to my artistic research. Still, *Data Proxy* is not real, but is rather a reference to the real. As such, it differentiates itself from tactical media (Apprich 2013), which are interested in building ›alternative‹ infrastructures and draw its users into the ›alternative‹ cultures and politics associated with it.<sup>17</sup>

Returning to the notion of media art, Quaranta rightly notes that in Germany, the discussion was dominated by MedienKunstNetz (Daniels, Frieling, and et. al. 2001), a major contribution to a historically rooted understanding of media art, including early modernist experiments and artists from the field of visual arts (Quaranta 2013, 23). Notable here is the difference between Media Art and the term New

---

<sup>17</sup> Another recent project, <https://adversarial.io> (2019–ongoing), where I collaborated with the hacker Flupke, is set up as a tactical media project, since users can have the real experience of altering image material so as to make it unreadable for machine vision (Neural Editors 2021).

Media Art, with the latter stressing the element of newness (as celebrated to some extent at the Zentrum für Kunst und Medien Karlsruhe and even more so at Ars Electronica, Linz). Media archeological art doesn't stress newness per se, but rather builds on the genealogical rootedness of infrastructural practices with media technology that is always new. Does this differ from Medienkunst as defined in Daniels/Frieling? The difference is slight, yet not negligible: While the original (European) Medienkunst concept draws on references to the arts and aesthetics of modernity, infrastructural media archeological art also includes references to the production and cooperation of that period. While Medienkunst may be perceived as a historically rooted project, infrastructural media archeological art rather investigates genealogies.

Medienkunst, a euphoric child of the 1980s and 1990s, grew into postmedia art, which is in a far sadder and more disillusioned state (Lovink 2019). Namely, hopes for the democratization of media usage have been shattered by the rise of ›social‹ networks which are filled with right-wing and hate speech and are oriented towards affective capital. They have replaced the previously cozy, utopian small-circle media experiments in capitalist sociality on a large scale.

*Data Proxy* navigates between these poles: 1.) Modernism-infused Medienkunst, 2.) post-modern, ahistorical postmedia art and 3.) an infrastructural, media archeological art approach. Following this discussion of the conceptual problems around Medienkunst, the next section addresses specific artistic strategies that inform *Data Proxy*.

### 1.2.2 Artistic continuities: overidentification and narration

*Data Proxy* is based on two strategies: overidentification and narration, which are revisited briefly here.

First, overidentification and appropriation: Overidentification criticizes through appropriating existing signs and codes, visual and oral languages, and driving them to their extremes. In the past, the artist collective *Neue Slowenische Kunst (Laibach, Irwin, Gledališče Sester Scipion Nasice)* has most notably looked into totalitarian aesthetics (Arns and Irwin 2003; Arns 2004). The overidentification strategy has been also used by others, for example, by Christoph Schlingensiefel in *Ausländer raus – Bitte liebt Österreich* exploring racism (Lilienthal and Philipp 2000) and by artists Eva and Franco Mattes of 0100101110101101.org in their installation *Nike Ground*, emphasizing the processes of gentrification (Mattes and Mattes 2003). More currently, overidentification has been updated by the postinternet collective DIS claiming to show »*The Present in Drag*«. Since 2013, they have run the stock image library *DisImages.com* (Akel 2013). My works have made use of the overidentification and appropriation approach in *The Setun Conspiracy* (2005) and in *History Exhaustion* (2008). Since the strategy of ›overidentification‹ has been explored elsewhere, notably in Inke Arns' Dissertation *Objects in the Mirror May Be Closer than They Appear* (Arns 2004, 91–107), I will not further elaborate it.

*Data Proxy* builds on the strategy of overidentification by appropriating the language and narrative strategy of some of the most successful vloggers. Appropriation has been chosen as the strategy for intervening into how capitalism has been able to co-opt all divergent, subcultural strategies into products of the culture industry. From a perspective where subcultural production no longer significantly hegemonizes the cultural field, the available option appears to be the appropriation and reinterpretation of existing cultural forms. This approach corresponds with Situationist artistic and political practices,

which showed how appropriation can be used to subvert what they called »the spectacle« (Ohrt 1990). Presenting the *Data Proxy* through the form of a vlog and in the figure of an influencer acknowledges the huge cultural impact which YouTube culture (in the sense of spectacle) has today.

Second, narration and dramatization: Narration and its potential for fictional scenarios has had a significant influence on my work. This is attributable in particular to Ilya and Emilia Kabakov's installation *Ten Characters*, for instance, which offered spectators the possibility of visiting the individual living rooms of 10 different characters. While it was only possible to experience the environments of the fictional inhabitants, the installation created a panorama of life in the late totalitarian Soviet Union of the 1980s (Kabakov and Grojs 1998). The Kabakovs' work is an exemplary demonstration of how the use of a protagonist makes it possible to dramatize history with artistic means.

Originally, the theater was the realm of dramatization. This is not the place to develop a deeper discussion of theatrical dramatization, but it is important to note that a common problem for the visual and performing arts relates to the narration of the invisible, of the systematic, of structure, and of infrastructure. Bertolt Brecht maintained, for example: »Petroleum resists five acts« (Brecht 1992, 303). The solution to such resistance in theatrical dramatization appears to be in the development of a protagonist, a subject who experiences and represents the non-depictable. Similarly the Kabakovs' introduce a figure into their *Ten Characters* to talk about abstract issues. For instance, *The Man Who Flew into Space from His Apartment*, one part of *Ten Characters*, deals with the loss of Utopia in the moment of its realization – in this case the Soviet space program –, and it not only deals with the loss of Utopia, but also its inevitable perversion.

Data Narration holds fictional potential and thus the possibility of critically rethinking the world as it is today through an infrastructural lens.<sup>18</sup> The next section will therefore explore how data can be narrated.

### 1.3 »What that means to you« – Towards the dramatization of data practices

Artist Memo Akten and architect Liam Young have proposed to use ›data dramatization‹ as opposed to ›data visualization‹ to critically discuss data practices: »Your motivation is not to tell the story of what happened in that particular instance, the story of the specific events that gave rise to that particular dataset. Your motivation is first to understand and absorb what *happened*, and then communicate what *is happening*. And more importantly, *what that means to you*. You create a fiction that is driven by those same processes that you have just discovered...« (Akten 2015). Akten's ›what that means to you‹ approach has been more recently applied by designers Cristina Cochior & Ruben van de Ven, who worked to »counter the grayness of datasets, to instead saturate them« (Cochior and van den Ven 2019). My research will take their approach further, abandoning the grid structure as a visual metaphor and concentrating instead on developing knowledge about data practices that is part of a general data literacy.<sup>19</sup> I would maintain, however, that *Data Proxy* is strongly influenced by the media

---

<sup>18</sup> To position my practice in media arts rather than in theatre is a matter of choice, as theatre and film are a highly hierarchical apparatus compared to the former.

<sup>19</sup> Data literacy is not seen as a pedagogical task which needs to be provided to the audience, but is rather understood as



archeological notions of ›data‹ as developed in my theoretical work, for instance the notion of transactional data in Episode 5 or with data production as a cultural technique (without ever designating it as such) in Episode 1. Resonating with the notion of data dramatization, *Data Proxy* has thus become a fiction representing the social processes which I and others have discovered. Since *Data Proxy* turns to vlogging as both metaphor and strategy, it makes sense to theoretically examine how narration in vlogs can overlap with data dramatization.

First, vlogging as a drama: A vlog, in order to remain interesting, does not only present condensed life. Vlogs build on the macro-narrative form of drama: losing weight, experiencing depression, falling in love, going through a breakup, et cetera. ›Macro‹, for the scale of narration, is rooted in the limited immediate resources of the actors in different genres such as van life, boat life, tiny house, pranks, travel or fashion.<sup>20</sup> Over time, in the instance where a vlogger successfully attracts attention and can capitalize on it, these resources may grow. Initially, they are rooted in the triad of economic, social and cultural capital as described by Bourdieu. The dramatic dimension here appears as social and cultural capital, which is turned into economical capital through audience labor, as discussed in *Audience Labour on Social Media – Learning from Sponsored Stories* (Fisher 2015) and *The Digital Subject – People as Data as Persons* (Goriunova 2019). Macro narration creates a need for the constant renewal of drama to keep the audience interested. Successful vloggers seem to have accumulated the necessary social capital to be able to deliver in this regard. Drama is also needed for my approach since *Data Proxy* doesn't opt for a talking head ›how to‹ or ›advice‹ column, but rather extended macro-narrative vlogging, which includes elements such as daily routines and personal flaws.

Second, vlogging as performative practice: Vlogging is a performative practice (Butler 1991, 202) of curating the self. Vlog practice intertwines with algorithmic figurations and systems of ›social media‹ platforms. Algorithmic figurations again co-curate the self, as media scholar Tyžlik-Carver points out: »In this new curatorial context, it is not just object/subject that is curated but its data. Often defined as traces of our digital lives data in fact is not a representation of every single user and her interactions with the World Wide Web. It should be rather considered as realization of correlations and situated connections among many agents that generate data. Content curating makes a person function in the regime of big data to select, manage, distil, and contextualize content while also becoming its essential part. Content curating is also part of computational counting and visualization, which makes it directly operational for big data« (Tyžlik-Carver 2017). Tyžlik-Carver argues here that every appearance in social media ecologies is already curated. It is curated by the user/creator in reaction to and in collaboration with »agents that generate data« (ibid.). Further building on Tyžlik-Carvers' argument, one can observe how the ›difference of subjectivities‹ translates into social capital (Bourdieu). While addressing and recreating normativity in countless van life, gardening, tech or make-up vlogs, it is the difference within the normative that keeps the audience watching. Difference as differential hook can be found in fluid gender roles, as for instance celebrated in *Coming Out, Elle Mills Style* (Mills 2017). Another example is the introduction of conspiracy theories when vlogger Logan Paul in *Alex Jones is...*

---

already existing common sense, which can be activated or triggered by addressing the subconscious typical experiences and data practices of everyday life.

<sup>20</sup> This notion of drama leans more strongly to a plot-driven side of dramatic narration and less to the antagonist practices of drama or flame war videos, as theorized in *Drama in the YouTube Community* (Pihlaja 2014).

*Alex Jones – Impulsive Ep. 60* invites right-wing conspiracy theorist Alex Jones, who had previously been banned from YouTube (L. Paul 2019).

Vlogs are mass media. They do not just thrive on consent. Mass media always calls for dissent (Luhmann 1996, 176 ff.) to keep a topic relevant. However, in contrast to journalism, in vlogs the difference needs to be executed and embodied by a subject. That subject again must be able to perform difference.

Third, the platform aspect of vlogging as drama: The vlog format is deeply intertwined with the platform. YouTube, TikTok and Snapchat are major platforms with their own respective data ecologies. Data ecologies include content guidelines, monetarization schemes and feedback practices such as likes, comments or views. Feedback practices turn into metrics and inform the creators about the effectiveness of their dramatization strategies.

Instead of deepening the discussion, I will focus briefly on *Data Proxy's* autoethnographic approach: On the one hand, it makes sense to become immersed in a specific platform data ecology in order to describe its embedded practices from within. On the other hand, the episodes of *Data Proxy* will not have the traction to create a data ecology that is comparable to high-value vlog channels. Thus, the evidence created is expected to be a long-tail experience with maybe several hundred views and a few comments per video. However, if the work were to accumulate a total of five-thousand views, that viewer count would equal that of an exhibition at a German Kunstverein.

*Data Proxy* is primarily produced for the YouTube ecosystem. It endeavors to reach a public that goes beyond the exhibition space. This harkens back to at least one of my previous projects, which was produced as a radio play. It was broadcast on WDR 3 and was simultaneously part of an installation where visitors encountered a large-scale sculpture and framed black pictures, both of which referred to the radio play.

Based on this experience, there should be an approach to combine both broadcast and exhibition audiences. Compared to radio, YouTube allows for additional interaction through its social media component. It should be noted that the development of a YouTube audience is a long-term project. According to professional YouTubers, it takes regular posting for at least one year to achieve recognition. Casey Neistat reiterated in his vlogs that his success began when he switched to daily postings. In short: The six episodes of the *Data Proxy* project will only provide a glimpse into YouTube's ecologies, since it is practically impossible to scale it towards the data ecology of large and successful vloggers.

An exhibition audience and YouTube viewers are different. Because of the difficulty of exhibiting under pandemic conditions, it is worth finding an alternative to exhibition spaces for distribution.

## 2 Vlogging on YouTube – How a platform shapes visual content

The discussion will now offer an analytical view of the platform ecology of YouTube and vlogging in the years leading up to 2020. The contemporary context significantly differs from what it was 10 years ago. YouTube is a maturing product, has clearly shaped formats for its respective audiences and mediates between advertisers' expectations, its own aspirations for extracting profit, creators' needs and audiences' entertainment needs. There have been earlier studies around these topics, and while they may provide historical context, articles from the 2010s appear to have aged, partly because of major techno-social changes around YouTube. So here is an update.

### 2.1 Vloggers' burnout and platform metrics

#### 2.1.1 Algorhythmia

Julia Alexander, a blogger and journalist, collected a list of YouTubers with psychological breakdowns and burnout. The list includes Bobby Burns (1 million subscribers), Elle Mills (1.2 million), Rubén ›El Rubius‹ Gundersen (30 million), Erik ›M3RKMUS1C‹ Phillips (4 million), Benjamin ›Crainer‹ Vestergaard (2.7 million), PewDiePie (62 million) and Jake Paul (15.2 million) (Alexander 2018).<sup>21</sup> One of them, Canadian YouTuber Elle Mills, addressed her success in the video *Burnt out at 19*: »This is all I ever wanted. And why the fuck am I so fucking unhappy? It - It doesn't make any sense. You know what I mean? Because, like, this is literally my fucking dream. And I'm fucking so unfucking happy. It doesn't make any fucking sense. It's so stupid. It is so stupid« (Mills 2018b, at 2:10). Mills points towards the realization that it was precisely the fulfillment of her desire for recognition, of being commercially successful and of being productive, that caused her to have a mental breakdown. So, where does the pressure come from? Is an unhealthy lifestyle with long working hours and a non-existent work-life balance the reason, as Alexander points out? Or are there are other factors at play as well?

One could argue that these incidents are not specific to YouTube and are common among artists, actors, politicians and other public figures. In contrast to previous generations of entertainers, YouTube creators' fears are directly linked to the production of data and, more generally speaking, to machinic agency. Previous Google employee Guillaume Chaslot, for instance, points out that YouTube optimizes its pattern recognition AI for watch time and catches viewers through its recommendation system (Chaslot 2017). YouTubers know this and try to act accordingly. Since the changes are not directly documented and sometimes sparsely communicate, creators need to A/B test and see what works and what doesn't.

The need to instantly react to algorithmic changes and overcome personal obstacles to productivity has been dubbed by media scholar Patricia G. Lange as Algorhythmia: »Effects of Algorhythmia include burnout, attempts to reverse engineer the algorithm, and an overall decrease in satisfaction among

---

<sup>21</sup> YouTubers are surprisingly vocal about the pressure. A keyword search for »truth about vanlife« returns 71 videos on YouTube, with 17 of them having more than 10,000 views, and up to 2 million views for the top-performing video. [https://www.youtube.com/results?search\\_query=truth+about+vanlife&sp=CAM%253D](https://www.youtube.com/results?search_query=truth+about+vanlife&sp=CAM%253D), accessed Sep 9, 2020.

viewers and creators regarding their media« (Lange 2020, 54). In this sense, vloggers are highly aware of their constantly exhibited, created, re-created and consumed subjectivity. They are aware of how the media impacts their lives. And they are subjected to how YouTube's algorithmic figuration plays into the anxieties of its creators: »Not knowing how YouTube's monetization system works, while also battling fears of videos being suppressed and less frequent uploads hindering their careers, are major anxieties« (Alexander 2018).

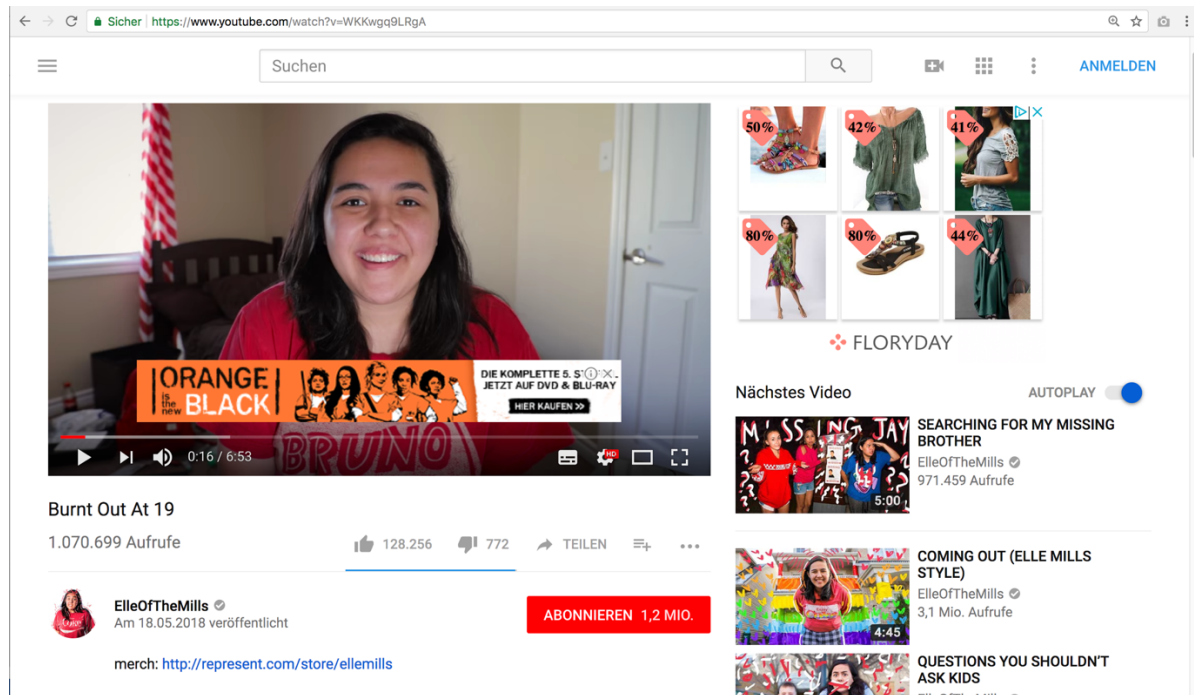


Fig. 6: Elle Mills on Youtube.com (Screenshot, June 2018).

»Perpetual beta«, the constant changing of code and social practices, has become the new norm. Practices around the YouTube platform get adapted, developed, and updated at all times.<sup>22</sup> Compare this with how often the practice of producing and reading books has changed, which could be counted in terms of centuries or decades. A look into YouTube's API updates<sup>23</sup> provides a glimpse into the frequency with which the platform modifies its black-box algorithmic figurations. Between January 2013 and December 2017, 66 updates were published, with each update introducing new features, taking away old functionality, correcting errors and updating existing functionality and algorithms. Following this breakneck rhythm of newness must be tiring.

<sup>22</sup> First conceptualized in: *Web 2.0 Principles and Best Practices* (Musser and O'Reilly 2007, 10). In her upcoming dissertation, Tatjana Seitz argues that perpetual beta has become »a very specific power condition within the multi-sided markets model that is specific to online platforms« (Seitz 2018, manuscript).

<sup>23</sup> The platform provides an application programming interface (API) through which the platform's data and functionality can be accessed from the outside via third-party software applications. Of course, the API is restricted according to the platform's own interests. From these restrictions follows a politics of data which limits data and algorithmic power for the end user, compared to what power the providing company has in terms of computational capital. An API thus is enabling and limiting, a tool of power and a hierarchically organized access point to the epistemic potential stored in databases. See *Objects of Intense Feeling – The Case of the Twitter API* (Bucher 2013).

All these perpetual beta changes resonate with what humans actually do on the platform. So, how does Algorhythmia then translate into human agency within the platform?

### 2.1.2 Agency and interpellation

When abstracting from a single case, it is possible to discover the specific nature of the hailing (Althusser) towards top creative vloggers through algorithmic agency. The platform encourages the provision of new content with intense frequency. It fosters the need to remain relevant, i.e., to provoke affective interactions such as comments, likes and clicks on advertisements and tracked purchases from online stores.<sup>24</sup>

If they do not resort to product placement or merchandise, YouTubers like Elle Mills earn their income through measured transactions and interactions. The specific quality of measurement is expressed through its density and continuity, in contrast to the TV ratings or box office success of an actor in traditional mass media. Density means that every possible action, reaction, interaction and transaction is recorded and valued. Continuity means that the recording reaches beyond the initial platform through the use of identification techniques such as cookies,<sup>25</sup> logins<sup>26</sup> and browser fingerprints<sup>27</sup> across various web spaces and devices. Facebook, for instance, embeds and makes it possible to embed its Facebook pixel, a small, invisible graphics file. The company describes the goals in the following terms: »Reach the right people. Find new customers, or people who have visited a specific page or taken a desired action on your website. Plus, create Lookalike Audiences to reach more people who are similar to your best customers« (Facebook for Business n.d.). Other objectives are: »Drive more sales«, and »Measure the results of your ads« (ibid.). Similar techniques are employed by other major companies like Google and its subsidiary DoubleClick.

While the measured transaction's continuity and density transformed into visible metrics can be described as hailing (Althusser) or interpellation (Poster), this operation is, for the individual creator, not straightforward. A single creator may get a general grasp of where s/he is situated through metrics. How transactional data is preprocessed (Gitelman 2013) is beyond his/her grasp. Whatever the metrics may convey, it is up to the creators to interpret the numbers. We realize that the assumed agency, through the availability of metrics, is limited by the creator's ability to ›read‹ statistics.

---

<sup>24</sup> For a critical examination of user creativity and productivity, see *The Digital Labour Theory of Value and Karl Marx in the Age of Facebook, YouTube, Twitter, and Weibo* (Fuchs 2015). The notion of affect used here follows *Affect* (Cifor In: Thylstrup 2021, 37–46).

<sup>25</sup> A cookie is a small text file saved on a person's device when visiting a certain website until it gets intentionally denied or deleted. Third-party cookies get installed through advertisement networks on devices that have visited a certain website, where the code of this website includes a reference to a particular advertisement network. These third-party cookies allow tracking of trans- and interactions across various web spaces and devices.

<sup>26</sup> A login occurs not only in the obvious way of entering a name and password. With the emergence of apps on mobile devices, where just one login is required during installation and on-boarding, it was made far less transparent if not even hidden, that each use constitutes a login.

<sup>27</sup> A browser fingerprint is identifying information which is transferred through the web browser to a particular website. Its traces remain even with changing IP addresses (when users change networks, for instance). The fingerprint combines originally distinct attributes, such as browser name and version, screen resolution and operating system of the device, plugins and installed fonts, with the last two being the most revealing (Laperdrix, Rudametkin, and Baudry 2016).

The need to create continually engaging content leads to a specific type of engagement creativity. It is expressed through the thin line walked by creators between an attractive title and thumbnail picture, on the one hand, and the viewers' dissatisfaction with clickbait, an unfulfilled promise, on the other. Specifically shaped vlog creativity also comprises affective storytelling, the ability to create pranks, to master virality and to deliver just the right amount of positivity adjusted for an audience. YouTube's engineers fine-tune the platform twice, towards hailing creators for continuously new content and towards hailing the users for maximum views. Viewers and creators react to these fine-grained changes. Video spam is another reaction to algorithmically-maximized viewing time. Artist and writer James Bridle pointed to videos such as *Surprise Play Doh Eggs Peppa Pig Stamper Cars Pocoyo Minecraft Smurfs Kinder Play Doh Sparkle Brilho* (Blu Toys Club Surprise 2016), and demonstrated the automatic and repeated creation of certain tropes in many variations. These videos are uploaded by specialized YouTube content creators simply because they get clicked and viewed – be it by children who can't fully comprehend what they are seeing or by click-fraud bots. The title is maximized for optimum search algorithm results, written in a gibberish language consisting of keywords. Bridle showed that the new quality of data labor lies in its microscopic units which get assembled, and only generate value when assembled: »And right now, right here, YouTube and Google are complicit in that system. The architecture they have built to extract the maximum revenue from online video is being hacked by persons unknown to abuse children, *perhaps not even deliberately*, but at a massive scale« (Bridle 2019). Video spam can be considered as the nemesis of algorithmically maximized viewing time.

### 2.1.3 Metrics, user interactions, and the production of value

So how do creators earn income? YouTube's partner program is coupled to its advertisement Platform Google AdSense, which again is closely integrated with the pervasive tracking service Google Analytics. As of 2018, it is estimated that Google Analytics is included in more than 73% of the top 100,000 most accessed websites (Rogers and Brewer 2018). In addition, Google Analytics or other tracking services are enabled for Internet-of-Things devices, e-mail newsletters, messaging platforms, smart phone apps and more.

Google Analytics then provides a platform for advertisers, auctioning off space on YouTube. Each embeddable video contains a clickable banner ad space. Another advertisement space is placed in YouTube's marginalia column. The AdSense program automatically places different advertisements each time pages reload. Another space for advertisers is pushed at the beginning of or within creator's videos. Video ads interrupt the original programming. In this ecology, a clickable URL for an advertisement contains the browser-/user-specific Google Click Identifier (GCLID), for instance: »gclid=EAIaIQobChMIidm9rOHS2wIVFD7gCh3J9wDZE AEYAS ACEgKLYvD\_BwE«. Google Click Identifier tracks users' movements from YouTube to the advertisers' websites, where again interaction within the website can be recorded. If a product is purchased, this transaction is registered as »conversion«. Conversion is the goal of the advertising company. The data aggregated through Google Click Identifier informs the advertiser about the success rate of a specific advertisement and provides the YouTube platform with indications of the value of the specific space where the ad was placed.<sup>28</sup>

---

<sup>28</sup> The full URL for this specific click by the author was »<https://www.floryday.com/de/Hemden-r9851/?currency=EUR&>

This ongoing, immediate production of metrics, the evaluation by others and the self-evaluation towards productive optimization are central to how machinic agency influences today's labor process. It is not only observed with web 2.0 platforms, but also in the field of Customer Relationship Management systems like Salesforce.com, where the actions of sales personnel are monitored and reported through numbers and diagrams, and to some extent also in quantified-self applications.

For Elle Mills, the pressure exerted by metrics led to what she described as burnout. Her carefully edited, seven-minute video *Burnt out at 19* received massive views. Wearing a red hoodie from her own merchandise line, Mills tells of her success story in short clips. She comments on it: »It's not what I expected. I'm constantly alarmed, constantly stressed, and always feel this overwhelming pressure« (Mills 2018b, at 1:49). Mills then describes a trip to London to visit friends, who helped her to cope with the situation but »were not able to cure me«. She then announces her absence to put her »mental health first« and ends with scenes from current and old video performances, where she sings a song together with her friends (ibid.). The professional editing, however, suggests that her leave of absence has been re-scripted after the fact.

About 7500 comments – that is, valuable interactions with the content – have been added by commenters. To a large extent, they comprise supportive messages, including personal advice based on commenters' own experiences. Almost 129,000 likes had been added by viewers to the video over the period of one month.

These interactions have been theorized by Schäfer/Mühleder, building on Boyd/Crawford (2012, 671), as three forms of connectivity. The first form, »articulated connectivity«, refers to more stable and visible connections between users, expressed by being a follower, exchanging email or keeping the phone number under contacts. The second form, »behavioral connectivity«, refers to patterns which can be observed from transaction data. »These might include people who text message one another, those who are tagged in photos together on Facebook, people who email one another, and people who are physically in the same space, at least according to their cell phone« (ibid.). These networks are relevant, for they lead to the additional distribution of a creator's content. Now Schäfer/Mühleder introduce a third form of connectivity, »algorithmic connectivity«, where they maintain the importance of a platform's ecosystem, which strives to keep users there. For YouTube they identify the auto-play function, which leads users from one video to the next, along with the recommendations, to be key agents of algorithmic connectivity (Schäfer and Mühleder 2020, 134f.). With this in mind, it is clear that these three modes of connectivity – articulated, behavioral, and algorithmic – create an affective space which is curated by YouTubers and viewers alike.<sup>29</sup> YouTube's ecosystem provides a dedicated place for situating affection by unifying all three forms of connectivity in one platform. At this point, the metrics kick in and convert affective user actions into value. Measurable actions, although microscopic from the single user's perspective, constitute labor and thus aggregate value for the advertisement space.<sup>30</sup>

---

[country\\_code=DE&utm\\_term=5264428&mid=1175963&not\\_pop=1&currency=EUR&country\\_code=DE&gclid=EAIaIQobChMIidm9rOHS2wIVFD7gCh3J9wDZEAfYASACEgKLYvD\\_BwE](#)«. The variable `utm_term` identifies the keywords that triggered the display of a certain advertisement.

<sup>29</sup> Affect was discussed in depth in *The Affect Theory Reader* (Gregg and Seigworth 2010).

<sup>30</sup> Audience labor was discussed in depth by (Fisher 2015; Nixon 2015).

The valuation of attention space is expressed by the platform through the price advertisers have to pay for placing an ad. I investigated the value of specific keywords according to Google's AdWords service, inferred from the conversations of the commenters. Placed on top of a web page, the keyword ›burn-out‹ costs the advertiser and earns Google 0.99 € while ›depression‹ is at 2.97 € for an international, English-speaking audience.<sup>31</sup> Advertisement revenue on YouTube through the Google AdSense program is split at 68% for the creator and 32% for Google (Google Inc. n.d.). The revenue is unnegotiated, since no global creator's union or global performing rights society exists to contest Google's conditions. Over the past years, the advertisement revenue for all of Google (incl. YouTube, Maps, Search, Gmail) rose from 95 million US dollars in 2017 to 134 million dollars in 2019 (Alphabet Inc. 2020, 29). The argument developed here is that the generally positive vlog of Elle Mills used the ›burnout‹ entry not only to appear more authentic, but also to create controversy and additional audience labor in the comments section. However, it is necessary to mention that the pressure on YouTube's top performers is actually high, since a recent study found that »a mere 0.42 percent of our total sample, account for 69.2 percent of subscribers and 62.4 percent of views« (Rieder, Coromina, and Matamoros-Fernández 2020). This means that once a YouTuber is in the top tier, the attention distributed is comparable to that of a mass media product, not an individual small enterprise.

## 2.2 Vlogging infrastructures

The discussion of metrics has already provided an initial glimpse into the underlying infrastructures of YouTube. In 2005–2012, YouTube ran on Linux, Apache 2.x, MySQL and Python, all of which are open-source technologies, and gradually added other Google technologies (some of which have been open-sourced or re-built by the open-source community) such as BigTable, the Google File System and the MySQL extension Vitess. In these early years Google reportedly just had an engineering team of two system administrators, two scalability software architects, two feature developers, two network engineers and one database architect, amounting to 9 total in 2008 (Hoff 2008). Although current numbers are difficult to find, there is an implication of growth, since today the company provides broader services such as YouTube Live, YouTube Kids and YouTube Premium. In 2012 the company still hadn't closed in on revenue to its competitors owing to high costs for the streaming and hosting of the video files. The company noticed that though it had data about the use of its services, it didn't have much knowledge about mobile use and embedded videos on third-party websites, which at that time were experiencing massive growth (Titlow 2015). Further technological development became driven by the need to monetize the content more intensively, to become profitable. A series of technological developments were introduced to address scalability issues. The add-on Vitess, developed in 2011 and open-sourced in 2015, added another layer to the existing and reliably functioning MySQL database that made it more distributable and scalable on a global level. Around 2015, YouTube discarded its original video search software, which had been developed and maintained by a small team, in favor of integrating the mightier Google Search engine. Part of this infrastructural update was the inclusion of machine learning algorithms and the removal of the original regression-based algorithm for video

---

<sup>31</sup> For comparison, the keyword ›bread‹ is worth 1.59 € internationally [June 14, 2018].



recommendations. »We demonstrated that using the age of the training example as an input feature removes an inherent bias towards the past [...]. This [...] increased the watch time dramatically on recently uploaded videos in A/B testing« (Covington, Adams, and Sargin 2016, 197). The promise was to increase watch hours for both content and advertisements. The then newly-installed neural network turned out to be better at grouping new video content to existing topical clusters (Titlow 2015). Google Analytics became more tightly integrated. A statistics API allowed access to video statistics from the outside, providing professional content creators with more incentives (or hailing) on how to best perform. The global delivery of videos, stored with the distributed Google File System in dedicated datacenters, is made possible through Content Delivery Network providers that enable further growth in streaming. Global video delivery implies the problem of scalability, of providing the right amount of network and data storage resources to a seemingly ever-growing amount of data. Holt and Vonderau discuss the infrastructural implications of data centers in general and of video streaming in particular. They offer the critique that: »While part of this redundant traffic is necessary to manage varying bandwidths, streaming delay, packet loss, or server failures [Sanjoy Paul 2011], a large part of it is indeed unnecessary, wasting network bandwidth and over-utilizing server-side resources [Yao Liu, Fei Li, Lei Guo et. al., 2013]. In short, seen from its user end, the ›cloud‹ looks like a pipeline plugged with often inessential and even progressively devalued data—or, in Marc Andrejevic’s words, like ›an environment of data glut‹ [Andrejevic 2013:41]« (Holt and Vonderau 2015, 86). Despite proof that streaming is more resource-intensive than traditional broadcast media like phone, TV or cable, to date the growth of streaming seems unstoppable.

Today, YouTube invests in upcoming video formats, new video compression and content formats such as Music, Live, Gaming and YouTube Originals. It is also looking into how it can grow in lower-bandwidth emerging markets. In 2017 YouTube CEO Susan Wojcicki announced that 10,000 new content moderators had been added to the team.<sup>32</sup>

This is this infrastructure that Elle Mills and her followers and commenters build on. It would be worth diving deeper into Mills’s universe to explore how she makes use of the infrastructure. For the sake of brevity, a small selection of her most successful videos’ titles serves to illustrate how she employs subjectivity in order to get measured and compensated, creating value of herself for herself and for the platform: *Slumber party with my brother and all his exes*, *Coming out (Elle Mills style)*, a video which went viral and became her commercial breakthrough, *Pranking my Dad’s ex-wife*, *Finding a stepdad on Tinder* and *My first kiss* were amongst the most popular. In Mills’s case the tension needed to attract attention is driven by pranks, confession and personal drama.

### 2.3 A survey of vlogging aesthetics

Now, after having establishing the demand for affective content, this survey investigates how this need expresses itself in the visual language of vlogging. This survey proposes two economic arguments. The first refers to production economics. Vloggers like Marques Brownlee and Elle Mills were initially solo freelancers who eventually grew into a company, often involving family and friends. To sustain the

---

<sup>32</sup> The difficult work of content moderators has been investigated in the documentary *The Cleaners* (2017).

operation, they need to constantly produce new content. The second argument refers to the need to create new visual content, according to YouTube's algorithmic demands. Since appearing in YouTube's recommendations drives views, the individual creator is invested in providing exciting content. To elucidate how these economics favor certain visual languages, this section surveys the following vlogs: Anna's Analysis (veganism, van life, soul searching), Marques Brownlee (tech), Elle Mills (pranks, LGBTQ), Casey Neistat (travel, entrepreneur), Kelly Stamps (fashion, minimalism), Sailing La Vagabonde (travel, family), Tara Brabazon (academic life) and Markiplier (gaming). I selected these vlogs for their success in terms of views, viewer engagement and public reception, for variety within mainstream content and for English-speakers. This approach alludes to a specific cultural sphere and excludes marginalized perspectives, remaining in the hegemonic sphere of cultural dominance of the liberal ›West‹. The major reason lies in my approach to appropriate and repeat these success strategies for my own project, the *Data Proxy*. Before diving into the material, the following paragraphs establish the methodology to be employed.

Vlog aesthetics contain two layers. One layer is the platform's Graphical User Interface (GUI), which in this case is YouTube. The GUI formats and organizes the visual ecology in which vlogs are embedded and entangled. Since there is already an abundance of a general critique and exploration of GUIs, it is not further explicated unless needed.<sup>33</sup> This documentation instead concentrates on the second layer, on how creators interact with and contribute to the visual ecology of YouTube. It therefore examines graphic design, titles and performance along with cinematographic features such as setup, frame composition, camera movement or visual frame<sup>34</sup>, mise-en-scène, editing and sound. It also looks at how YouTubers perform in the visual frame. One aspect, the interaction of vloggers with their viewers through comments or platform-external communication, is omitted here and requires further research at another point.

1.) Vlog aesthetics: Apparently, not much scholarly material on the aesthetics of vlogging exists. Most articles which refer to vlogging are written through the lens of the social sciences or from a marketing perspective.<sup>35</sup> Only one study, *Evoking presence in vlogging – A case study of U. K. beauty blogger Zoe Sugg* (Zhang 2017), looked into questions of image composition. The study mainly concentrated on composition and excluded aspects of montage or visual language. Having established the need for more

---

<sup>33</sup> See, for instance, the journal *Interface Critique* (Heidelberg University Library), or the discussion about dark patterns in user interface design: *The Dark (Patterns) Side of UX Design* (Gray et al. 2018), *Dark Patterns at Scale – Findings from a Crawl of 11K Shopping Websites* (Mathur et al. 2019).

<sup>34</sup> As proposed by Zappavigna, ›visual frame‹ stands in for any actual technology which has been used to capture an image (Zappavigna 2019, 6).

<sup>35</sup> *The YouTube Reader* (Snickars and Vonderau 2009), *Rants, Reactions, and other Rhetorics – Genres of the YouTube Vlog* (E. A. Werner 2012), *Talking back in cyberspace – self-love, hair care, and counter narratives in Black adolescent girls' YouTube vlogs* (Phelps-Ward and Laura 2016), *Watching people playing games – A survey of presentational techniques in most popular game-vlogs* (Pietruszka 2016), *The Stylistics of Selected American, Italian and Polish Challenge Vlogs* (Kurpiel 2017), *Crying on YouTube – Vlogs, self-exposure and the productivity of negative affect* (Berryman and Kavka 2018), *Consumerism, Sociability, and Techno-optimism? An Ethical Analysis of the Promotion of Social Norms in YouTuber Zoella's Vlogs* (Schelenz 2018), *Wolves in Sheep's Clothing? Influencer Marketing beyond Endorsements – Deconstructing Persuasive and Questionable Marketing Strategies in Vlogs on YouTube based on Social Influence Heuristics Framework* (Rohde and Mau 2019), *The Organised Self and Lifestyle Minimalism – Multimodal Deixis and Point of View in Decluttering Vlogs on YouTube* (Zappavigna 2019), *Algorhythmia* (Lange 2020), *Konnektiver Zynismus und Neue Rechte – Das Beispiel des YouTubers Adlersson* (Schäfer and Mühleder 2020).

research, it is necessary to mention that, for reasons of brevity, the following analysis concentrates on the current state of 2020 and thus differs from genealogically grounded approaches. While vlogging on YouTube in 2010 could mostly be described as vernacular or DIY aesthetics, today's bloggers have matured in terms of filming and editing practices.

This chapter then looks from within the material itself (hours of watching vlogs), instead of using established visual categories, for example, from cinema studies. It nonetheless borrows from cinema studies to establish descriptive categories. Therefore, it uses terminology such as frame, shot, camera distance, camera angle, camera movement, shot duration, cut, fade, wipe, transition and montage sequence (Villarejo 36–49). For brevity, however, the section will not look deeper into the relation of image and sound, although a discussion of the relation of original sound versus post-production sound effects and music would be fruitful.

2.) Performance aesthetics: Two concepts are applied to describe the performative aspect of vlogging. The first concept of performance by Erving Goffman has been developed from a sociological perspective, rooted in everyday life and business operations. The second concept of performance by Erika Fischer-Lichte originates in theatre studies, and thus in the performing arts. The entanglement of both approaches makes it possible to cover the performance of vlogging, which is positioned between business and entertainment (art).

Goffman looks into how make-believe and authenticity function from the performer's perspective: »At one extreme, we find that the performer can be fully taken in by his own act; he can be sincerely convinced that the impression of reality which he stages is the real reality ... At the other extreme, we find that the performer may not be taken in at all by his own routine« (Goffman 1956, 10). Goffman refers to the latter as the cynical performer. Obviously these two, the convinced and the cynical performers, are the two poles between which vloggers act. To mediate between the two extremes, vloggers make use of a professional »front« (ibid., 13), which is described here as the channel's visual appearance and the scenery used in the videos. A successful YouTuber is thus a person who is able to mediate between ›setting‹, ›appearance‹ and ›manner‹. Instead of giving a full overview of Goffman's performance theory here, the focus will instead be on the question: What are the idealized performances that the vlogger-performers intentionally offer to viewers/users? How do they convey performance? What can be expected is that these differ slightly, depending on the professional self-image that each vlogger performs, be it gamer, traveler, entertainer, entrepreneur, fashionista or tech enthusiast. Goffman's perspective does not cover the entertainment aspect of performance in vlogs, for which we need to turn to another theoretical field.

Theatre studies has discussed the shift from dramatic theatre towards artistic performance. Although such a shift may not describe the entertainment aspect of vlogging closely enough, it provides a preliminary concept for vlogs as performance and performing in vlogs. Theatricality refers to how a lecture, a talk or a vlog is performed and staged, and also to the impression it leaves on the viewer. Theatricality suggests how the performer is able to impersonate a specific habitus (Bourdieu), which marks them as having a certain persona, e.g., a professor, a traveler, a tech journalist and so on. How does theatricality appear in vlogs?

The difference between theatre play and theatrical performance is identified by Fischer-Lichte as being that of an actor closely following the dramatic text in classical theatre and that of the actor's use of the body in all its expressivity in artistic performances of the 1960s. Vloggers, like contemporary actors, synthesize these two poles of ›representation‹ and ›presence‹. Representation finds its place when

vloggers become stage characters and follow a text or script, an »authoritative controlling mechanism« (Fischer Lichte 2008:147). Presence refers to immediacy, opulence, and authenticity. Presence is exerted when vloggers talk off-script, when they put their bodies at risk, as in pranks and stunts, when they act excessively. For the viewer, performative presence generates meaning that can trigger »chains of association« (ibid. 148), while dramatical representation generates meaning that constitutes character. Both types can be observed when vloggers act. For vloggers who communicate knowledge, such as Anna of Anna's Analysis, Tara Brabazon or Marques Brownlee, performance can be read two-fold: It acknowledges the integrity of the person and it acknowledges the degree to which the person is knowledgeable. Vloggers regularly break the fourth wall and speak to their viewers out of character. It happens so frequently that it has become a habit.

Performance further stretches beyond the integration of a single channel and creator into YouTube's larger visual ecology. YouTubers and viewers, for instance, interact with each other through comments, and they perform (Fischer-Lichte) or curate (Tyzlik-Carver 2017) their presence on the platform. So, the performance doesn't just happen within the video, but rather emerges outside the video.

In conclusion, it is important to address what methodology has been employed, and which differentiations have guided the research. The tasks are to describe 1.) the visual language including typography and style, 2.) the performance of the creators 3.) cinematographic qualities, including composition, montage, setup and sound and 4.) how they integrate with the larger visual ecology of the YouTube platform.

Since this research is qualitative and not quantitative, the approach has been to divert the research along the category with the largest difference: Among the reviewed vlogs, two major formats could be identified regarding the setup. The first format, ›talk‹, consists mainly of a talking head situation, where the viewer is addressed through the fourth wall in a relatively static setup like a studio or a living space. The second format is ›cinematography oriented‹, often serving as an interlude or parenthesis for the talking head sections. It is marked by changes of scenery inside and outside. The two setups show the largest differences and constitute the topic of the next sections.

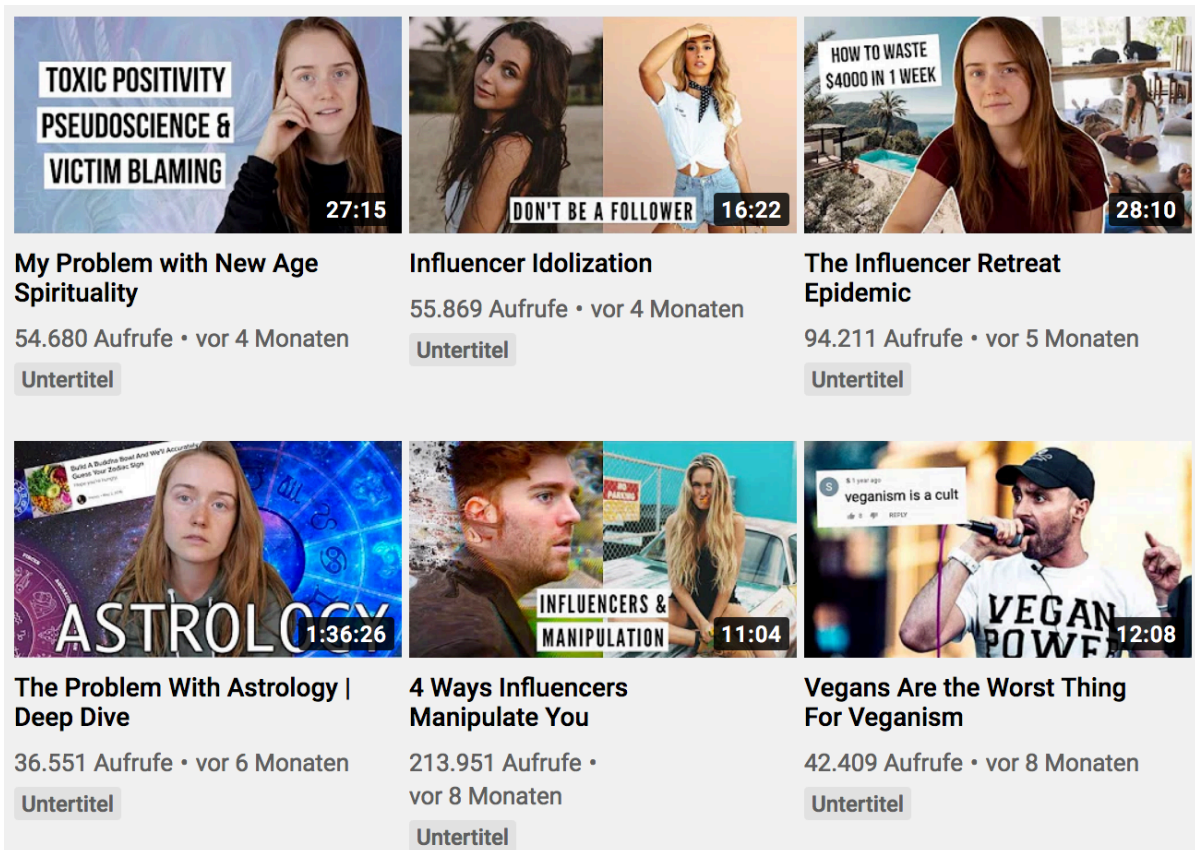


Fig. 7: Selection of YouTube videos by Anna's Analysis (screenshot, author).

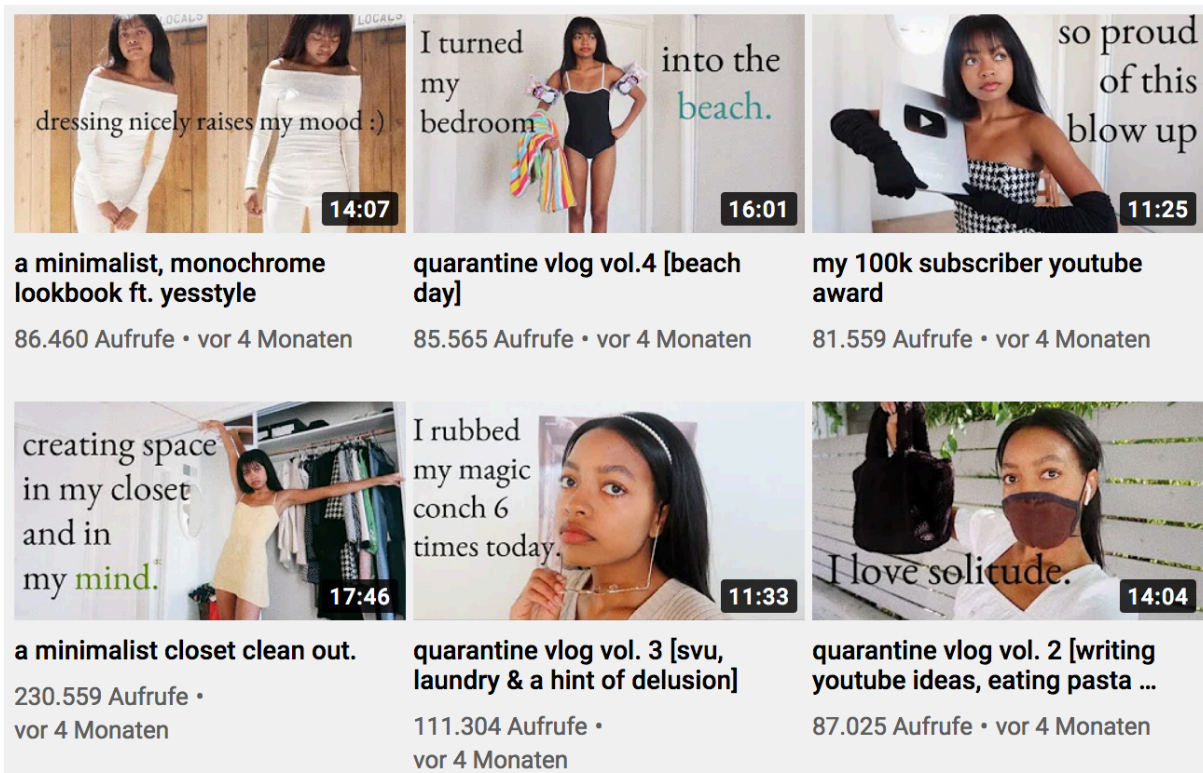


Fig. 8: Selection of YouTube videos by Kelly Stamps (screenshot, author).

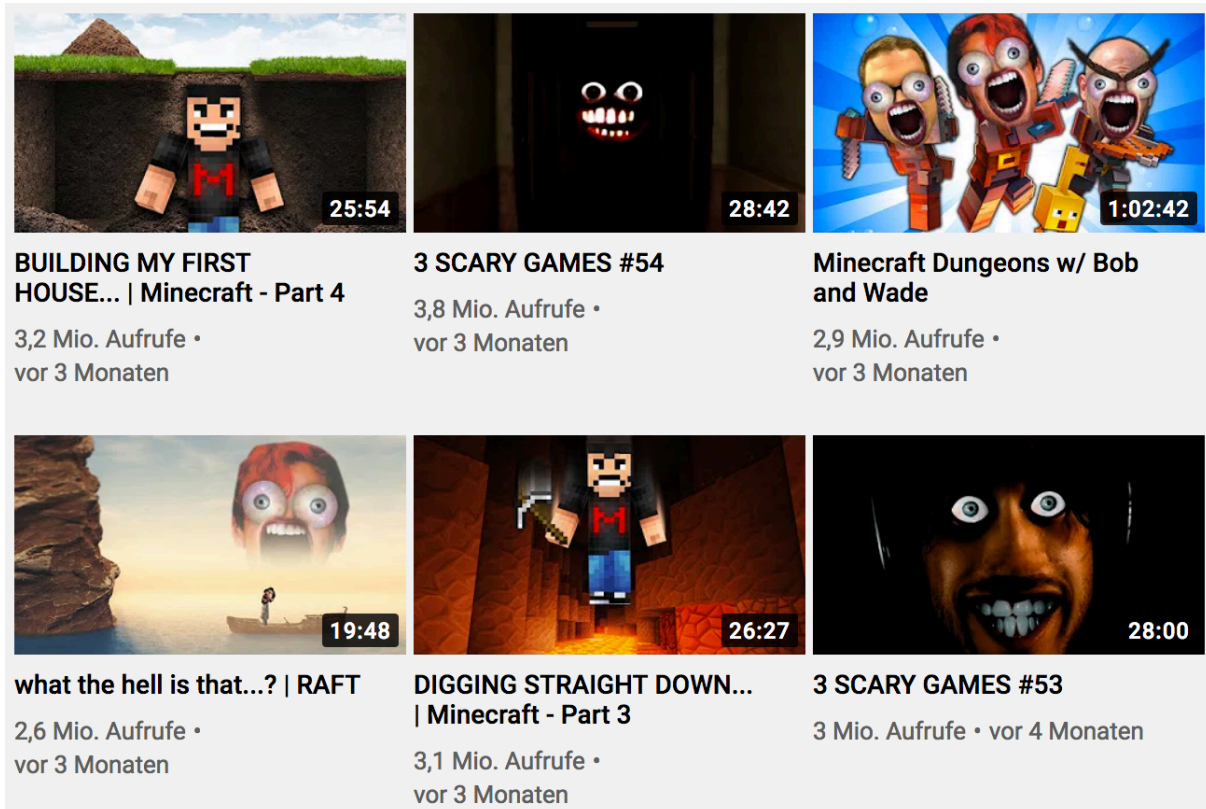


Fig. 9: Selection of YouTube videos by Markiplier (screenshot, author).

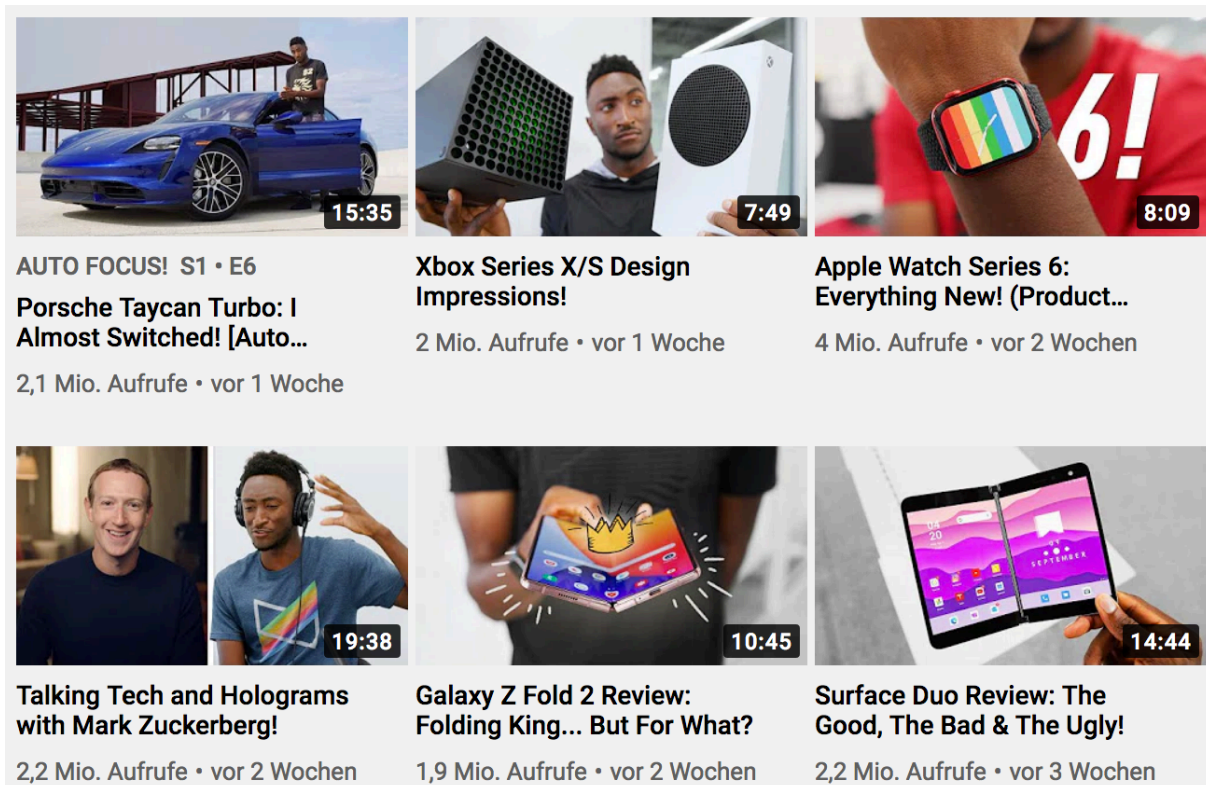


Fig. 10: Selection of YouTube videos by Marques Brownlee (screenshot, author).

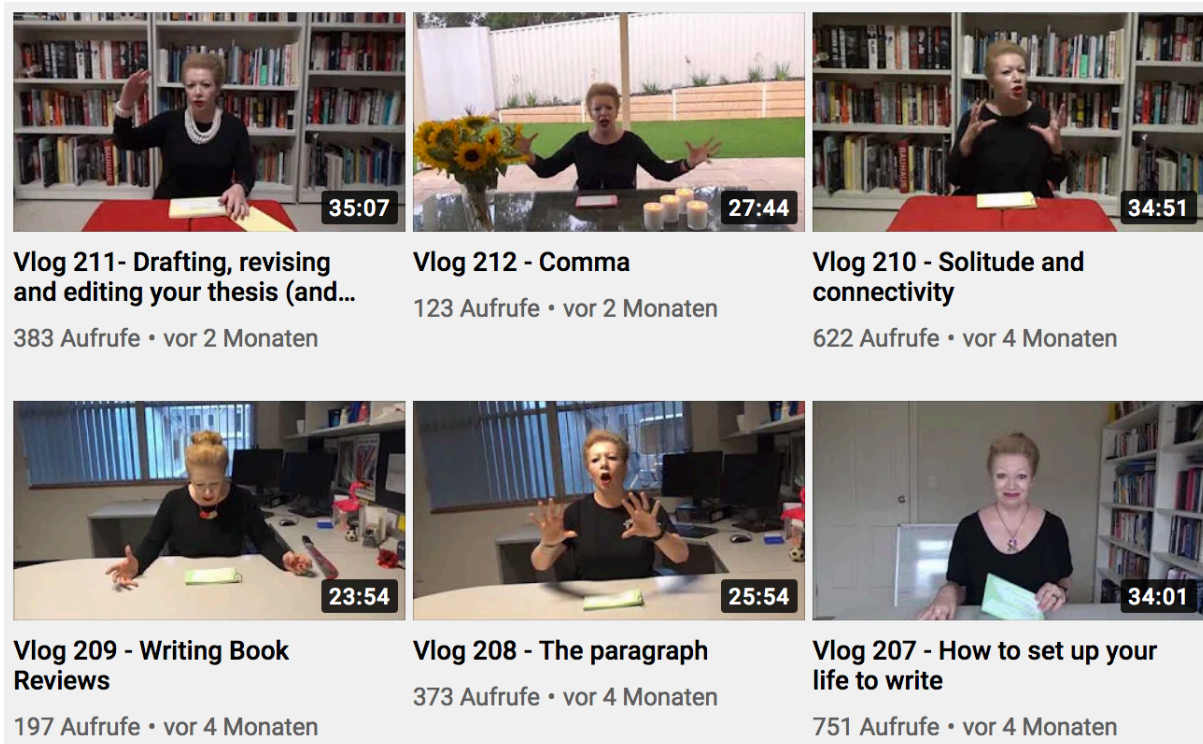


Fig. 11: Selection of YouTube videos by Tara Brabazon (screenshot, author).

### 2.3.1 Talk in a static setup

Of the vlogs under discussion, Anna's Analysis, Marques Brownlee, Kelly Stamps, Tara Brabazon and Markiplier fall primarily into the ›talk‹ category. Marques Brownlee deviates slightly from this group, since he has diversified his formats and uses interviews, reportage, tech reviews in exterior spaces and educational film formats in addition to ›talk‹.

#### 2.3.1.1 Visual Language incl. Typography and Style

An important element for attracting viewers is the use of video icons and titles. Anna's Analysis and Kelly Stamps use text and typography in their preview icons, while their peers in this group don't. Different levels of media professionalism can be observed:

Academic Tara Brabazon uses thumbnails from raw video footage showing her exploring an issue, often with great intensity. Her open mouth implicates that she is talking and her arms and hands are wide open in an engaged, scholarly gesture. The preview icons show her either in her office space, or since the pandemic in her home office, outside in the yard, or inside, in front of a blackboard. These unedited preview icons deliver a certain rawness and immediacy while contributing to an authentic picture. Her titles include a running number followed by a short sequence, as for example, *Vlog 204 – The sentence*, *Vlog 206 – Fast writing*, or *Vlog 210 – Solitude and connectivity*.

Fashion vlogger Kelly Stamps also uses shots from her videos, although they appear edited for brightness and contrast and contain typography. In serif letters, she delivers messages like ›dressing nicely

raises my mood, creating space in my closet and in my mind, I love solitude«, which differ from the actual video titles: *a minimalist, monochrome lookbook ft. yesstyle*, *a minimalist closet cleaned out* and *quarantine vlog vol. 2*. Thus, she uses the video icons to communicate additional hints about the content. The icons show her only and nobody else, since the vlog revolves around her alone. Her darker hair and skin color is set in contrast to the bright, often white backgrounds. Of the sample, she has the most diverse range of body poses, which refer to her use of K-pop, cheerleader, modelling and vogue style performance within the videos.

The preview icons for Anna's Analysis are visibly edited, so that, for instance, her body is set apart from the background by a thin white line. They often feature black and white sans-serif titles which complement the content of the videos. These icon titles, e.g., *How to waste \$4000 in one week*, are different from the actual video's titles, in this case: *The influencer retreat epidemic*. Half of the preview icons show her in portrait frame, looking with a calm expression at the viewer. The other half features images of people or gear, which she associates with the vlog's main message. Motifs include other influencers, famous vegans whom Anna criticizes, or an esoteric water filter, which she critically researches. Her visual strategy is somewhat similar to that of tech enthusiast Marques Brownlee, although he doesn't use typography often. His preview icons are strongly edited and have a professional, ultra-clean, stock-photography industry look, similar to the tech objects which he reviews. About half of the preview icons show his face, often with an inquiring, skeptical expression. The other half concentrates on a specific tech object such as an Apple Watch, Galaxy Z Fold 2, and so on, which get presented as a visual fetish. When an object is depicted from a point-of-view perspective, it is held by hand, with the implication that Brownlee is using or interacting with the object. In one instance, an interview, the picture is divided, with one half showing him in an engaged posture and the other half featuring his interlocutor, Mark Zuckerberg. His video's titles always end with an exclamation mark or a question mark. The gamer Markiplier features comic-like game graphics. They often show a character from a specific game, sometimes with Markiplier's face. Comparable to other gamers, he uses a playful, expressive style, based on the corporate identity of specific games which get integrated into the channel's visual language. In one instance, the preview icon diverts from this scheme and shows Markiplier's portrait, signaling that it is a personal message (*The problem we need to confront*). Most of the titles refer directly to a computer game, for instance, *BUILDING MY FIRST HOUSE | Minecraft part 4*, or *That can't be good | RAFT*. These titles make an effort to be controversial and attention-grabbing, unlike Kelly Stamps's or Tara Brabazon's rather dry video titles.

### 2.3.1.2 Performance

Tara Brabazon introduces female professorial performance with a hint of theatricality.<sup>36</sup> Brabazon's theatricality thus shines light on the other vloggers' performances and their ability to assume a persona – such as traveler, gamer, parent, tech journalist and so on.

»Greetings to you, my name is Tara Brabazon, and I'm the dean of graduate research at Flinders University and welcome to vlog 228, »Productivity (during tough times)« (Brabazon 2020). Brabazon's first sentence establishes the subsequent 56 minutes of academic performance. Sitting behind a desk, Brabazon talks straight to the camera. She introduces the example of Isaac Newton who during the bubonic

---

<sup>36</sup> Regarding professorial performance, see the anthology *Der Auftritt* (Etzemüller 2019), specifically page 17–23.



plague allegedly came up with his calculus and continues to talk about the myths of productivity. Speaking from notes, she uses her hands expressively to claim the space. She moves her hands towards the camera when addressing the viewership. Brabazon also spreads her fingers to take up even more space. Then again when figuring out how to proceed, she directs her gaze away from the camera, looking sideways, only to drive home her point by looking straight back at the camera, opening her eyes wide. She makes a comment about Isaac Newton and Robert Hooke. After giving a wink with her left eye, she marks the end of a sentence with her eyes and mouth wide open, and her thumbs and index fingers spread to create an imaginative frame, just to say, her voice now a whisper: »only joking« (ibid., at 1:29). It can be assumed that this academic performance was shaped and learned over the course of numerous lectures, teaching sessions, consultations and committee sessions. Therefore, Brabazon's variety of theatricality is not specific to vlogging. It is, however, able to produce a presence in a way that ensures Brabazon's academic credibility when moving from the lecture hall into another medium: a platform-enabled video service.

Anna's Analysis is involved with the analysis of reality and everyday life. It provides an interesting example of non-institutional citizen science, or what some call ›YouTube University‹. However, using the notion of ›YouTube University‹ assumes a tone that is a mockery of formal academic education. The distancing may be grounded in the observation of how platforms have fostered pseudo-science, conspiracy theory and fake news. Instead of lazily following this true but one-sided observation, the citizen science element should not be underestimated, since ›YouTube University‹ indeed makes it possible to break down institutional barriers of access to knowledge. Anna Reid establishes a positive, though not overly cheerful, character who speaks at a fast pace. She has her sentences prepared in writing, memorizes them and speaks them to the cam, using jump-cuts to edit out any mistakes. She appears calm and concentrated, which is underscored by her sparing use of facial expressions and hand gestures, which many other YouTubers use far more liberally. Sometimes, however, when she is about to make a point, Reid naturally uses hand movements to emphasize the importance and it seems that her usage of gestures has increased over the years. Compared to Brabazon, Anna's Analysis is on the opposite end of the theatricality spectrum, which shows that exalted and expressive speaking is not a prerequisite for attracting a viewership. Reid usually sits central in the visual frame and does not move around in the space. Her videos do not start with a formulaic greeting, but she instead asks the question which is the theme of the video, for instance »What would you do with complete time, financial and location freedom?« (Reid 2020b). She uses either screenshots or presentation slides when quoting directly or introduces third-party videography to underscore her arguments.

She shares this calmness of presentation with Marques Brownlee, who uses relatively more hand gestures and strong facial expressions while generally maintaining a calm speaking tone. His video *Galaxy Z Fold 2 Review: Folding King... But For What?* opens with his formal »What's up? MKBHD here« and then immediately dives into the topic. Relative to others, Brownlee uses more dramatic breaks in the flow of his talk, where he looks away and thinks about what to say, to then look directly into the camera. When reviewing a product, the imagery often shows the reviewed product from a point-of-view perspective, and thus disembodies the speaker, who becomes invisible and proceeds as a meta voice-over to the imagery.

In the case of Kelly Stamps, we notice that her talking head approach broadens the field by introducing aspects of performance. When announcing in one of her videos that »I'm not Stamps anymore, I evolved to Kelly Envelope«, she plays with words, puns and humor. Her comedic style infuses parts of

her videos, be it complaints, hauls or musings about life in general. Stamps divides her time between speaking to the camera while sitting in her chair and while walking around her room, which serves as her stage. The mostly static shots emphasize the theatrical expressiveness of her K-pop and cheerleading inspired poses, rhythmic gymnastics, figure-skating dance moves, facial expressions, and talking. She uses a lot of hand movements to point to her own body parts or to objects, such as recently-acquired articles of clothing, or just to gesticulate freely. While other YouTubers use gesticulation as a rhetorical element and part of speech, Stamps additionally employs ornamental posing gestures, expressive yet controlled.

These relate to her general play with the male gaze. The concept of the male gaze was introduced by Laura Mulvey for determining how the objectification of women in cinema works (Mulvey 1975).<sup>37</sup> According to Mulvey, the male gaze frames actresses as passive objects of male desire, making the cis-male viewer effectively a voyeur. Regardless of to what degree the vlogging genre in general is voyeuristic, Stamps indeed decided to own the situation and drive views and success by intentionally providing gaze(s) on her body. Using a pointing stick in relation to a set of poses in *how I save money at 23* (Stamps 2020a) she turns »the gaze on the gazers« (Soloway 2016) by playfully shifting between being-gazed-at and dominant poses and introducing herself as »Professor Kelly Stamps« (ibid.). Stamps seems to master the fine line of complicity with and deconstruction of the male gaze.

Markiplier alternates between looking directly into the camera, addressing the viewer or watching his computer screen, while narrating and commenting when playing a game. We can't see him gesticulating, but from time to time his hands appear in the frame when he interrupts the game, stops using the keyboard and talks to the camera. Markiplier uses a maximum of theatrical performance and expressivity through vocal and facial expressions, which compensate for the static frame. He screams at the screen during intensive gameplay situations, rolls his eyes, raises his eyebrows and makes humorous out-of-character comments, sings and screams »Oh my god« again and again (Markiplier 2020c). Originally, Markiplier was planning to create a comical sketch channel on YouTube. When the channel instead turned out as a gaming channel, he began to participate in comedy and improvisational theatre projects with his friends, which appeared on YouTube and on stage under the name Cyndago. He has managed to integrate these aspirations into his Markiplier character, which comes across as noisy, manly and sometimes funny.

2.3.1.3 Cinematographic qualities, incl. composition, scenery, montage, and sound  
Tara Brabazon provides the most straightforward vlog experience. Sitting behind a table with notes and recorded as a single mid-shot, she usually speaks for about 20 minutes without any cuts or changes in scenery. Imagery is shown by the existing light and is not color corrected. Sometimes the frame is not completely leveled and the simple camera and objective which Brabazon uses tend to add lens distortion. When she records the video in her office, her face is on the same plane as the frame, but

---

<sup>37</sup> It is critical to understand that Mulvey's original concept was developed for the non-interactive and apparatus driven setup of cinema production and for narrative film of the 1970s. The classic film set - large, complicated, hierarchical and logistically sophisticated - is different from the more intimate, small, partly DIY set of vloggers. When, for example, Mulvey describes »the conscious aim being always to eliminate intrusive camera presence and prevent a distancing awareness in the audience«, it becomes obvious that this is different for vlogging. Vloggers constantly discuss the production apparatus and make it part of their performance. Furthermore, the concept of »female gaze« has been discussed more recently (c.f. Soloway 2016).

sometimes she films from a slightly higher angle, especially in other places. The cinematography is thus amateurish and authentic. It demonstrates the origins of vlogging: straight, plain recordings of people speaking with whatever resources happen to be at hand.

Anna's Analysis uses two kinds of setups: 1.) a direct, talking-to-cam approach and 2.) the usage of third-party excerpts to illustrate the monologue, which then becomes voice-over. The first setup, talking-to-cam, shows her mostly in her living space, seated and talking in a medium close-up to the cam. Later in the video there are well-lit, longer sequences without cuts, although barely noticeable jump-cuts appear every so often to edit out any bloopers.

Departing from this, her video *The Reality of Van Life on Social Media* (Reid 2019b) consists entirely of a montage of excerpts from other YouTubers. At the beginning, the text states that these clips are being used according to ›fair use‹ policy. For each clip the creator's screen name is shown in the lower left-hand corner. Her visual setup moves the focus towards Reid's calm, but fast-paced voice which partly goes along with or contradicts the imagery. Other videos, like *4 Ways Influencers Manipulate You* (Reid 2020a) take a combined approach, and cut together the talking-to-cam approach with excerpts from other YouTubers. The video also includes screenshots from the comments section as evidence for Reid's arguments. In ›Fake Positivity on Social Media (Reid 2019a), she completely uses voice-over narrating scenes of herself, acting and performing in her flat. These scenes are cut together through straight cuts, and do not follow a specific sequence.

Kelly Stamps's comedic style and daily life vlogs also work with different setups. It could be argued that when she is exploring Manhattan or looking for a new apartment, her vlogs are cinematographic and more dynamic. She is moving around outside, taking different b-roll shots of her surroundings before addressing the viewers. Stamps either holds the camera in selfie style or has it positioned on a tripod. However, in her apartment, the surroundings are more static and the time is usually divided between a performance part and a talking-to-cam part. One such example is *my packages keep getting stolen bc I live in the cut* (Stamps 2020b). It starts with Stamps standing in her room, with a white background in an almost full shot frame. The scene cuts after only two seconds to a zoomed-in mid-shot of her looking into the cam and displaying the words ›I'm about to go in‹, then cutting back to the previous shot. She begins to speak with ›Now...‹ and for a few seconds again the video cuts to close-up as she says, ›I won't say the name‹, and again cut, returning to the previous frame, ›but a particular delivery courier has been playing with my emotions‹ (ibid.). In a variation, she also includes the zooming in, adding even more movement to the close-up.<sup>38</sup> Stamps uses these cut-ins as a regularly recurring feature to create intimacy and urgency. Cut-ins restructure the otherwise static talking-to-cam approach, more than the occasional changes of camera angle, and have become a signature to her videos. In contrast, the usage of b-roll could not be observed, and almost everything that Stamps shows has an immediate reason or effect.

Marques Brownlee's cinematography employs many different formats. Besides his interviews with tech leaders like Bill Gates, Elon Musk and Mark Zuckerberg, he has a retro-tech format, a cutting-edge tech format and more. His most regular format is a tech review, which can be either situated outside (for cars, etc.) or inside. The outside scenes are, of course, more dynamic, with Brownlee moving around.

---

<sup>38</sup> I remember that during media art studies in a documentary class these ›cheap‹ zoom-in's were frowned upon as ›unprofessional‹.

For brevity, this section only looks at his indoor tech reviews, which he usually delivers while sitting at a table. The filming, frames, lighting and sound are flawless. Brownlee takes pride in an uber-professional, immaculate appearance. Brownlee doesn't experiment with frame or camera angle or cutting techniques. The only barely noticeable specific decision is having a slightly lowered camera angle. In *On the iPhone 12 With No Charger* (Brownlee 2020) for example, he uses the same type of cut-in-zoom as does Kelly Stamps, but to a comparatively minor affective effect. Jump cuts within a section of talk exist to hide bloopers. Unpacking sequences or handling a tech gadget may be provided with additional music and can be filmed from different camera angles, for instance, from above in extreme close-up, or from a point-of-view perspective. As Branigan has pointed out, »The POV shot is a shot in which the camera assumes the position of a subject in order to show us what the subject sees« (Branigan 1984, 103). It creates a common perspective shared by the viewer and the creator. Usually Brownlee's hands are still in frame, holding an object, just as a weapon is held by first-person shooters. Zappavigna argues that this perspective, especially when dealing with products, creates a specific fetishization of the object: »Factoring in the ›look with me‹ point of view of the videos, what is construed is a particular neoliberal subject position, that might be loosely glossed as ›if you see how I relate to my objects; you see who I am and who I want to become« (Zappavigna 2019, 11). While there is also a fetishistic dimension to non-POV presentations of products, in POV the subjectivity of the creator and of the viewer are conflated, shaping the fetishization effects to create a seemingly authentic experience. Brownlee's lighting is always perfect and creates an uber-bright tech atmosphere, a Cartesian clean, prototypically male-encoded space of thought and analysis, which Brownlee employs to validate his content.

Markiplier seems destined to sit in a chair and play games, while the visual frame shows his reactions. The first few seconds of each video show him full-screen, before the game itself blends in, and then his face is scaled down and shown frame-in-frame, usually located at the top left corner of the screen. However, in *EVERYTHING IS FINE. | Minecraft - Part 17* (Markiplier 2020b) Markiplier at some point yells »breaaaaaad!« and his yelling is intensified through the momentary enlargement of his frame (within the overall frame) and the addition of a sound effect to his voice (ibid., 18:03 min). When his character is involved in intense combat, Markiplier begins yelling and the complete frame is overlaid with flames, as his rotated frame (within the frame) shakes dramatically (ibid., 16:44 min). Most of the time, while being filmed from a tilted, high camera angle, Markiplier watches his computer screen, until at some point the game is interrupted and he addresses the viewers directly, breaking the fourth wall. At some point he talks with his video editor and sidekick LixianTV, who appears as an animated cartoon figure, sharing insecurities about the video's content (ibid., at 22:51 min). This sequence is shown again full screen, with the animated figure moving from right into the frame. Slightly different from the game videos are other formats where he discusses personal matters such as *Someone Wrote a Book About Me... WITHOUT MY PERMISSION?!* (Markiplier 2020a). Markiplier talks to the cam for 18 minutes, sitting in his room. The frame shows a medium close-up. *Someone Wrote a Book About Me* is strongly edited. It uses cut-zoom-ins, sound effects, animation, sub-titling and other edits in order to make it more dynamic.

With these descriptions of talk in a static setup in mind, this text proceeds to the next section and describes the cinematography oriented in a dynamic setup.

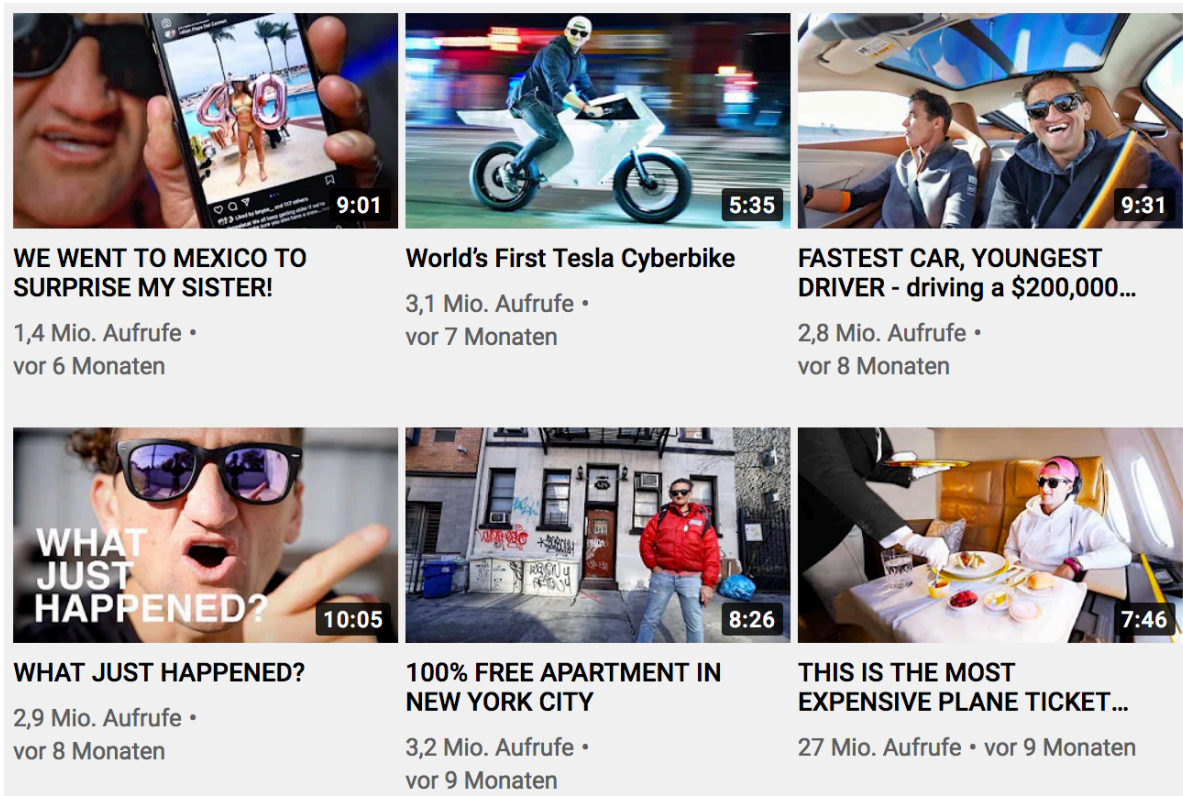


Fig. 12: Selection of YouTube videos by Casey Neistat (screenshot, author).

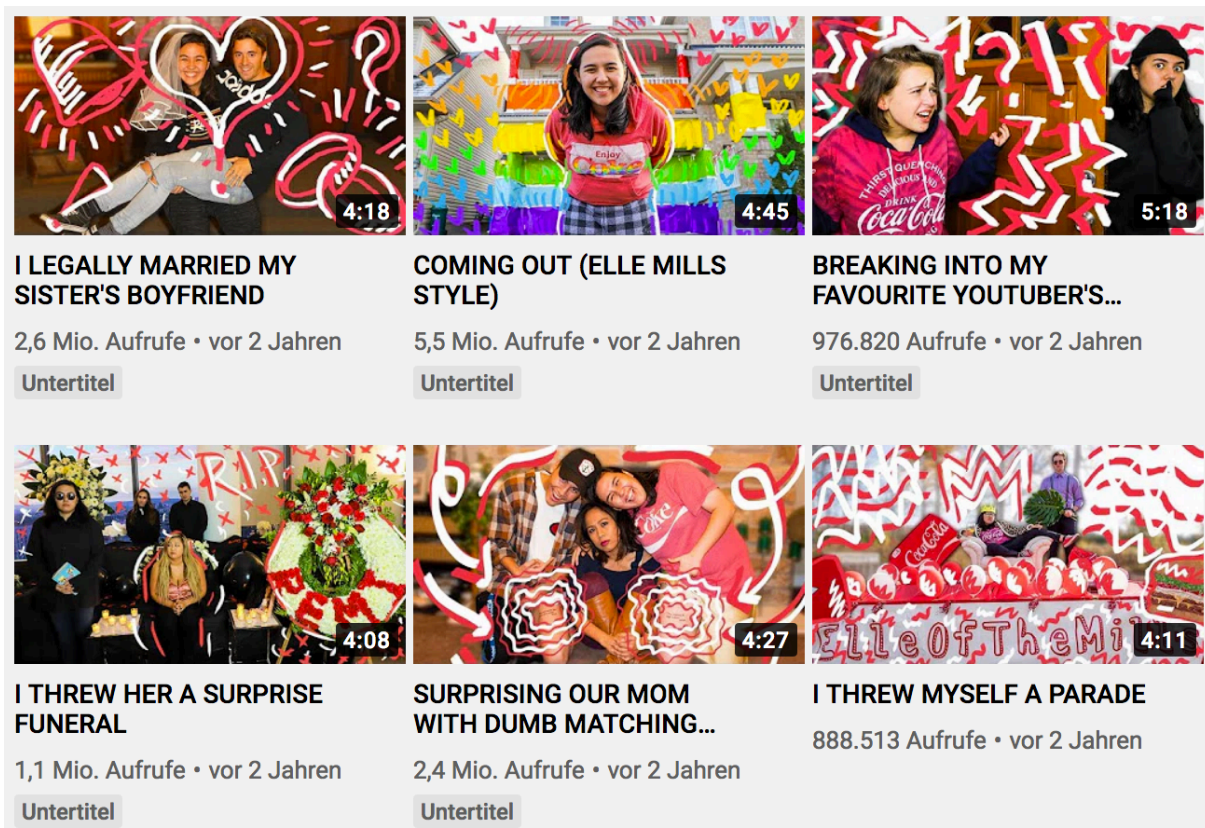


Fig. 13: Selection of YouTube videos by Elle Mills (screenshot, author).

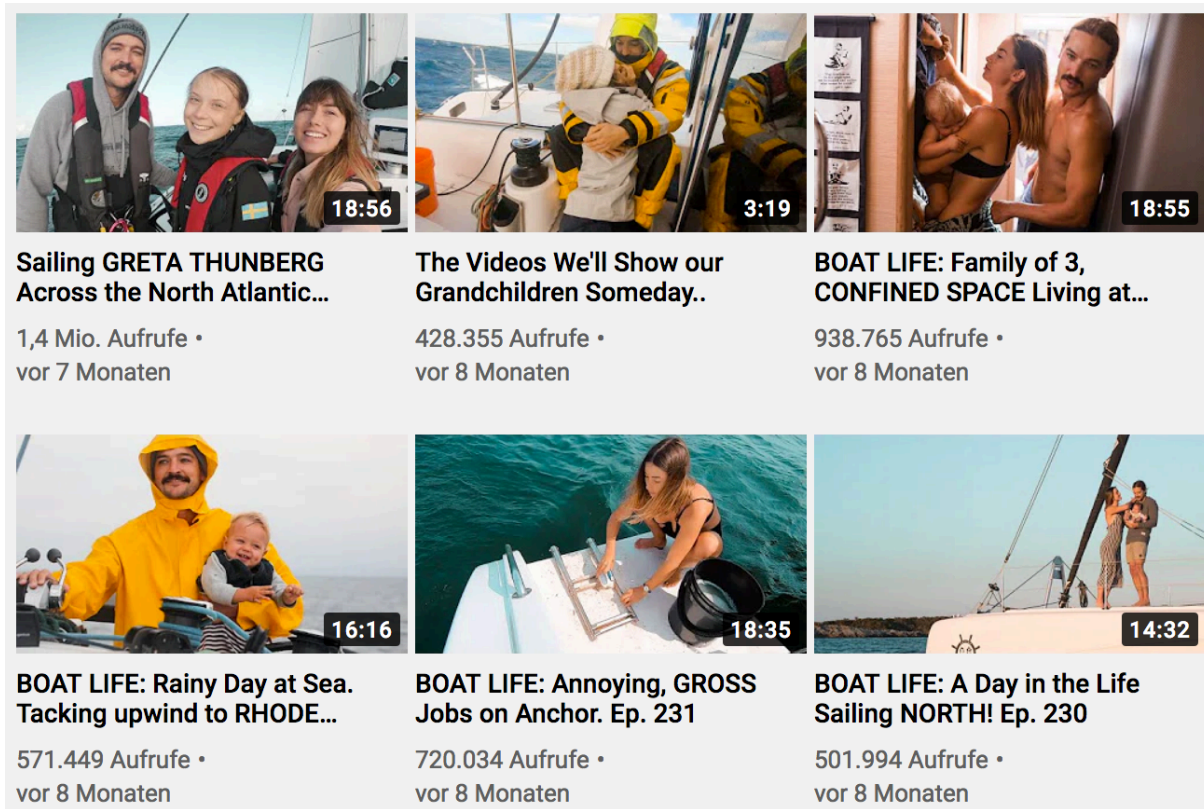


Fig. 14: Selection of YouTube videos by Sailing La Vagabonde (screenshot, author).

### 2.3.2 Cinematography oriented in a dynamic setup

Of the reviewed vlogs, Elle Mills, Casey Neistat and Sailing La Vagabonde fall primarily into the ›cinematography oriented‹ category, although all of them include significant sections which employ the talking head style. It must therefore be stressed that this is a subgenre of ›talking heads in a static setup‹. The majority of the observations made above apply to this section, since talking head situations are deeply integrated into cinematographic parts.

#### 2.3.2.1 Visual Language incl. Typography and Style

The video preview icons of prankster Elle Mills show her interacting with family or friends. Often, she is smiling or has a questioning or exalted look on her face. Strong red and white illustrative patterns are laid over these images and produce a hectic, excited and funny atmosphere, which complements her pranks and shenanigans. These graphics and merchandise are designed by Mills' cousin Sj Prejola, while another friend, Nick Le, provides the photographs for social media and the YouTube thumbnails. Titles in capital letters add to the visual excitement. *COMING OUT (ELLE MILLS STYLE)*, *I THREW MYSELF A PARADE* and *MY SCHOOL HATES ME, SO I PRANKED THEM (COPS CALLED)* are rather long titles, compared to others in the sample.

Entrepreneur and traveler Casey Neistat also uses capital letters, but in a different way: *I MESSED UP majorly in Abu Dhabi*, *WHAT JUST HAPPENED?* and *Model X is all messed up* are some examples.

These titles attempt to be controversial and interpellate viewers with attention-grabbing claims that border on clickbait. In this case, video preview titles seem to be screenshots and always show Neistat with his signature black sunglasses. While some of the preview icons show the protagonist staring calmly into the camera, others are action-oriented and involve more gesticulating. Neistat uses preset fonts, mostly Helvetica, in white against a moving image background, or just black.

Travel vloggers Sailing La Vagabonde present a lot of colorful, outdoor boat action taken from video footage. Usually, the preview icons feature two or three protagonists, creating interest in what is actually going on. Often a person is smiling. Until recently, no typography was used in these pictures. The descriptive titles either lead to a specific traveling experience, like *Sailing GRETA THUNBERG across the Atlantic*, or announce a domestic housekeeping-style episode such as *BOAT LIFE: Annoying GROSS Jobs on anchor*. Differentiation appears here as part of expectation management, which acknowledges that some episodes are less exciting in content than others. Nonetheless, they deliver a relaxed travel mood, and this specific subgenre is well-understood by the audience, as the comments show.

#### 2.3.2.2 Performance

In terms of performance, Sailing La Vagabonde are all smiles in their videos. There is a lot of joking, poking and laughs. One signature laugh by protagonist Elayna Carasu has made it into the trailer, which plays at the beginning of each episode. Both Carasu and Riley Whitelum are naturals in front of their camera, with their toned bodies and good looks, and the cuteness of their two-year-old son Lenny works as well. It may have helped that Carasu, before becoming a full-time YouTuber, was a diver, van-lifer, singer and entertainer. So, the performance is natural and all positive vibes. Possible bad vibes do not get included, apart from a few dramatic episodes, such as when Whitelum injures his neck and Carasu has to finish an ocean crossing (Carasu and Whitelum 2018a). How Carasu's and Whitelum's performance is tuned in to the family and travel sailing format became most visible when Greta Thunberg, a media professional in her own right, and her father Svante Thunberg joined La Vagabonde for an Atlantic crossing in mid-November 2019. First, irritated by the camera and constant filming, over the course of a few episodes, we see them loosening up, until they finally manage to reply »Good« and »Top of the world, as always«, when asked, how they are doing (Carasu and Whitelum 2020, at 13:01). Still, the overall constructedness becomes obvious to the viewer when in other episodes some scenes which weren't caught on camera are reenacted, or when out of the boredom of quarantine, a comedic episode of aerobics on the boat appears.

In Elle Mill's *I Learned a Foreign Language in 7 Days* (Mills 2020), she performs the somewhat awkward twentysomething. Speaking to the cam, she is focused, sometimes raising her eyebrows, questioning, then again smiling from time to time, but not overly expressive. In another episode, she is more clearly performing, which leads to a stronger expressivity in her face and gestures. What makes her acting appear natural is that she often works with sidekicks, friends, her brother or her mother. But she also knows how to deliver a punchline, taking a brief pause before announcing something funny. Over the years, Mills has grown from pulling funny pranks to delivering a consistent story and with that her acting skills have clearly grown. She has no reservations about performing, as when in the video *Meeting My Brother's New Girlfriend for the First Time* she catches an imaginary microphone to sing a pop song (Mills 2019, at 1:08). It is in this same video that she has her brother's new girlfriend read the following sentence aloud: »I promise to let Elle exploit our relationship for YouTube views« (ibid., at 5:43).

2.3.2.3 Cinematographic qualities, including composition, scenery, montage, sound  
How novel and formative Casey Neistat's cinematic style must have been is illustrated by a 2009 assessment by media scholars Burgess and Green, who placed the mid-millennium vlogging on the ›amateur‹ side of the content spectrum on YouTube (Burgess and Green 2009, 93). This was the time before Neistat, when vlogging meant a linear documentation from the beginning of the day to the end, primarily in selfie mode. Being an autodidact from the outset, and thus representative of many YouTubers' careers, Neistat was an important agent in the emergence of the cinematic style of vlogging. He introduced – or at least this is what he claims – nonlinear narrative structures into vlogging. Neistat breaks it down into a ›three-act narrative: beginning, middle, end; setup, conflict, resolution‹ (Neistat 2018, at 5:00). Originally, Neistat started out by putting his stop-motion animations and personal short films online on YouTube. These short films were either reportage-style, funny pranks or short explanatory films, which used stop-trick animation that borrowed from information design. It is noteworthy that visually Neistat owes just as much to the artist Tom Sachs, whom Neistat worked for when he was younger, as he owes to cinema in general. His usage of handwriting, his studio setup and even the desk which makes it possible to film stop motion resemble Sachs's.<sup>39</sup>

This aesthetic was further cultivated when, in 2015, he decided to deliver a daily vlog. The self-imposed challenge was to shoot a 10-minute video each day, beyond a simple talking head setup. He aimed to deliver a variation on the shaky video, by having handheld, as well as tripod-fixed shots, mixed with establishing shots, which would orient the viewer (ibid.). In an instruction to a video editor assistant, he wrote in 2020: ›I don't like filters, color correction, transitions, digital effects, fancy font treatment or titles ... use nothing but straight cuts and footage out of the camera, text is almost always Helvetica ... because it is the default‹ (Neistat 2020). Neistat's own ›Dogma95‹ + 35 manifesto is thus rooted in a period when ›fancy‹ plugins and effects were not yet the standard for smart phone video apps. It is an argument for the pureness and authenticity of the genre. Overall, the Neistat approach introduces a comparatively rough visual language which distinguishes itself from the artificiality of high-quality productions. It allows him and other YouTubers to continually provide street credibility to their fanbase, even if their rise in social status suggests otherwise. In this regard I agree with my colleague Gabriel Ben Moshe, who in a discussion pointed out similarities with the HipHop-Genre and the rise and fall of HipHop stars compared to their street credibility.

We can argue that this journey from amateurism to professionalism, in the sense of conformity to YouTube's algorithmic ecology, is paradigmatic for many YouTubers. The 2010s, however, have seen an infusion of professionally-trained editors, camera operators, and graphic designers coming to work behind the scenes.<sup>40</sup> There are already plenty of popular articles and videos which attempt to dissect Neistat's style and make it available for other aspiring vloggers; therefore, for the sake of brevity, only one example will be discussed here. Neistat's major visual argument is movement. He walks, skates, bikes, uses scooters, drives, rides by train or taxi to the city, he flies drones and by drone, travels by airplane, as well as occasionally by helicopter. Constant spatial change shapes his narration and provides an ongoing dynamic. In *THE ONLY THING BETTER THAN A BOOSTED BOARD*, we see

---

<sup>39</sup> Compare Tom Sachs's studio tour (Scholl 2019).

<sup>40</sup> Neistat has' cooperated with cinematographers Jack Coyne, Oscar Boyson, Tim Kellner, Dan Mace and others.



him coming out of a plane (selfie view), grabbing a cab to Manhattan (drone flight), riding an electric boosted skateboard (selfie view and point-of-view) and so on. Even when Neistat talks to the cam, he moves in and out the frame by walking towards the camera and talking in a close-up (Neistat 2016). In the second part of this video, static scenes of talking are cut against scenes of riding the boosted board. The riding scenes contain some of Neistat's signature music, which is cut away when he talks (ibid., from 7:15).<sup>41</sup>

A similar dynamic of moving around shapes the narration in *Sailing La Vagabonde. CLOSE CALL with a TIGER SHARK in the Bahamas!!!* (Carasu and Whitelum 2018b) starts with a drone shot of the boat's aft and sharks circling in the blue water (after having been lured with fish by the protagonists). The scene cuts over to an underwater shot of a shark attacking a diver, before the intro trailer sets in. Having delivered on the promise of the title and created another promise that viewers will see more exciting footage, the episode can begin. *Sailing La Vagabonde's* montage, edited by Carasu, mostly works with simple cuts between shots and jump cuts when talking to the cam, and less fancy transitions compared to Neistat. Plenty of b-roll, that is, shots of the boat's interiors, sails, coffee being prepared, the signature winch winding a sail rope, the boat gliding through the water, sunbathing on the deck and so on creates a nautical atmosphere of travel. Now the dialogue between Whitelum and visiting friends about shark observation sets in, filmed by Carasu. The conversation is framed in mid-medium shots, cutting between the talk, underwater footage, with talk continuing as voice-over, back to Whitelum or his friend Dan as they talk. Much of the subsequent underwater filmed action is unified by indie guitar music, promoting an eerie atmosphere. The sequence again cuts to Carasu talking in selfie mode to the cam about her observations when she shortly joined in the diving. Later, as they continue to discuss their experience, the frame switches again between the guys talking, until at some point, the frame changes, meaning that Carasu's camera has moved to another position, closer to Whitelum. It shows him in a medium close-up before zooming out to include Dan responding. The apparently effortless scene demonstrates the advanced filming and editing techniques on the part of Carasu, who is very aware of how to capture the scene.

Elle Mills's videos follow a three-step logic: an intro, setup and an event. In *My First Kiss* (Mills 2018a) for example, she begins with a self-made kissing booth, where she claims that Valentine's Day is her favorite day, but she has spent previous years alone. With this premise, the video continues towards the setup. The outdoor kissing booth scene changes into an indoor living room, where jump cuts show the protagonists in different outfits, ready for a date. Then, shown in a time lapse and without any speaking, Mills puts up posters reading »WANTED VALENTINE'S DAY KISS«. In tightly cut scenes, she shows girlfriends whom she could imagine dating and giving her a Valentine's Day kiss. These are straight, basic cuts. The frame gets split at this point. The larger frame shows Mills's friends on a Skype call. Mills asks them frame in frame if they want to be her Valentine's Day kiss. It becomes obvious that this part is scripted when we hear their absurd excuses. Finally, one of them, fellow YouTuber Corinna Kopf, agrees. The next sequence returns to full frame. It fast cuts to Mill's male friends consulting her on how to get the kiss done, then to a short interview between Mills and her potential kisser Kopf, and we see different scenes for preparing the stage. Against background music, several scenes, mostly shaky

---

<sup>41</sup> Neistat has been notorious in using and pushing a sub-genre of instrumental HipHop dubbed as Boom Bap or Chillhop, which he obtained for free from SoundCloud artists such as Dyalla, Floppy Circus, David Cuttur, Jeff Kale, LiamLRY, maxzwell, SAFAKASH and Maxwell Young, earning him the SoundCloud hashtag #neistat.

point-of-view, create an intense reportage style atmosphere, with the protagonists sharing their ideas. Now the event itself occurs. When they kiss, the music tunes in again and the camera goes into slow motion as glitter falls. Then, after having delivered, Mills stands in the same kissing both as in the beginning, going full circle. Task completed.

Mills edits her videos herself over the course of 12 to 36 hours. Her videos include a dense montage, including b-roll, and there are no fancy transitions. The argument is based not on fanciness, but on awkwardness, premise, execution and witty delivery. The videos are underscored with quality music, that is, her videos contain songs like *Pretty Woman* by Roy Orbison, for which she must have cleared the copyrights before republishing them on YouTube.

### 2.3.3 Conclusion

Two major approaches which constitute a specific visual language for vlogging can be identified. The first approach is rooted in the arm length that distances the cam from the creator and in a specific selfie culture of the genre.<sup>42</sup> The second approach is indebted to the ongoing digitalization of productive means: Whereas prior to the 2010s a helicopter or plane had to be rented, drones became available and, with them, an affordable bird's eye perspective. Where earlier time-lapse or slow-motion shoots required special camera equipment, time and space concerning aesthetic functions now come built-in into consumer products. Vloggers often take pride in the equipment they use and post (affiliate) links below each video. The equipment currently used includes:

- the action-oriented small GoPro digital camera
- point-and-shoot cameras like the Sony ZX-1
- smart phone with gimbal or tripod to steady the picture, smart phone sticks to extend the arm length
- drones, remote-operated
- high-quality DSLR cameras like the Canon D80
- LED lights, tripods, directed microphones

With these productive means in mind, we can identify which specific visual techniques of vlogging get used in selfie culture and the aestheticization of time and space on a platform like YouTube.



Fig. 15: Neistat's video *Nintendo Switch on an airplane* (Neistat 2017) is entirely constructed of swish pan transitions. Neistat attributes the swish pan to travel vlogger Justin Johnson.

---

<sup>42</sup> See *I Love My Selfie* (Stavans 2017), *Perceptions of selfie takers versus selfie stick users: Exploring personality and social attraction differences* (Bevan 2017), and *Exploring the Selfie* (Eckel, Ruchatz, and Wirth 2018).

1.) Selfie culture: The ›swish pan transition‹ can be achieved by moving the camera in one direction before a cut, for instance, from left to right. The next shot after the cut, after a change of scenery, begins with the same left to right movement. Both shots taken together produce a seamless change of scene. A variation is ›cover-the-lens‹, when a video's protagonist moves quickly towards the camera, covering the lens. The cut is followed by another scene, where the protagonist removes the cover from the lens, thus producing a dynamic of movement which transitions from one scene to the next. A variation of this technique is the ›object transition‹. It pans in on an object, cuts, changes scenery while moving the object, starts filming again and pans out from the object. Here the transition focuses on an object, transferring the viewer from one boundary object to the next. These transitions have an immediate mediatic quality for vlogging because they depend on the use of the outstretched arm. The ›selfie panorama‹ is another vlog-specific aesthetic strategy. Here the protagonists use a camera mounted on a selfie stick, extending his/her arm length. With the camera pointed at the protagonist's face, which constitutes the frame's center, the protagonist revolves around him- or herself, creating a circular pan. The selfie panorama pan is not primarily intended to concentrate on the face. It rather serves to show the environment, such as a beautiful landscape. Both Neistat and Sailing La Vagabonde make regular use of this arm-length pan-panorama. It adds tension through synchronicity: conveying visual pleasure in the background and relaying information in the foreground. This vlogging-specific technique only makes sense in a selfie-stick setup.



Fig. 16: Drone flight sequence. The camera moves seamlessly from the ocean towards the harbor and the city (Carasú and Whitelum 2017).

In Sailing La Vagabonde, the ›point of view‹ shot takes on another meaning, compared to Brownlee's product fetishizations. The non-mounted or non-fixed camera becomes a dynamic element of reportage, when Elayna asks Riley how spear fishing went. Point of view can transform into a ›back and forth pan‹, where the cam naturally pans between the interviewer and the interviewee. ›Back and forth pan‹, a swift change between selfie and point-of-view, is specific to the vlogging genre, since it involves the arm and the cam, which determine the frame, and both become part of the cinema apparatus.

2.) Aestheticization of time and space: Neistat and Sailing la Vagabonde have both invested in using drone shots. Without discussing drone shots in depth, it is worth noting that the visual rhetoric of the distanced overview in drone shots is clearly rooted in military surveillance operations and operational

images.<sup>43</sup> As media theoretician Lisa Parks points out: »a feminist critique of verticality not only calls attention to power's vertical operations and materializations, but, in the process, weaves in an analysis of the politics of difference and conditions of embodiment on earth« (Parks 2018, 24). How can this strong suggestion become productive for the usage of drone footage in vlogs? The vertical dominant seeing-without-being-seen creates visual pleasure which is not interrupted by any negative or disruptive events that could be happening ›down below‹ and therefore blends in with a mediated ›hygge‹ perception of reality. Beautiful, sublime landscapes and distant soothed cityscapes create a relaxed atmosphere. This is not, however, the only visual function of drone coverage, which is not necessarily distant. Neistat, for instance, also uses more close-up imagery of Manhattan's urban canyons, in an exploratory dynamic, that moves using fly-through mode. These close-up drone flights become part of his general ›on the move‹ rhetoric.

Even if the democratization of productive means has made drone shots widely available, they still involve their own set of logistics. At one point, Neistat got a warning note from the air traffic authorities for his drone flights over Manhattan. Such logistics include getting permissions as well as transporting and using the equipment properly. Vlog viewers value the extra effort which is spent by the creators and regularly comment on it below the videos.

Time lapses and slow-motion effects lead to similar observations. ›Time lapses‹ have a history of being overused and became despised by creators, because their narrative value is low. However, the compression of time becomes a device of aestheticization and a visual argument in three instances: First, it can show ›progress‹, for instance, when something is being built and documented on video. However, since cutting is a long-established montage technique to suspend time, the time-lapse compression is not really needed. Second, a time-lapse can compress longer events like a sunset. It can thus compress the aesthetic delight and make it more easily consumable. Third, time-lapses create timely overview, when showing a busy street crossing, such as Times Square in Manhattan, or people playing basketball. Such a particular reading accounts for compressed aesthetics of activity and business. At some point in the mid-2010s YouTubers discovered they could combine time-lapse with a pan using an extra 360-degree tilt pan device. These sequences synthesize the spatial overview of a circular pan, with the timely overview of a time-lapses, thus maximizing the aesthetic potential.

›Slow motion‹, popularized by GoPro action cams, is regularly used to underline the beauty of the situation. It slows down movement and keeps the viewer in the moment. Sailing La Vagabonde uses slow motion, for instance, when exploring a new beach, when enjoying nature or a beautiful city. Slow motion appears as an effect between normally-paced sequences. Used too continuously, it would become boring. It is the difference of speed in the montage that makes the aesthetic difference.

Both Sailing La Vagabonde and Casey Neistat have used drone flights combined with slow motion. Viewers then see a normal bird's eye perspective which flies from the sea to the shore, slowing down

---

<sup>43</sup> Drones and the aerial view have been discussed in the context of art through Harun Farocki's film *Bilder der Welt und Inschrift des Krieges* (1988), Josh Begley's *Dronestream* (2002-2017), James Bridle's *Dronestagram* (2012) and by curator Honor Harger in *Unmanned Aerial Ecologies* (Harger 2014). In media theory, drones have been researched in *Drone geographies* (Gregory 2014), *Flächen/Rastern. Zur Bildlichkeit der Drohne* (Andreas 2015), in the anthology *Life in the Age of Drone Warfare* (Parks and Kaplan 2017) and more recently at the conference *The aesthetics of drone warfare – War and art in the information age* (Humanities Research Institute, University of Sheffield, 7-8 February 2020).

where the water meets land and shows waves, beach and a shore, before speeding up in land again. These are strong visual arguments which dynamize space and time.



Fig. 17: Hygge style – calm, relaxed, and pastel (Reid 2019b)



Fig. 18: Attention-grabbing style – rich, colorful videography (Mills 2018a).

Aesthetic styles of vlogging are as broad as the topics and genres. They borrow from documentary, reportage, entertainment, auto-ethnography, and cinematic approaches. It is possible to identify three major visual strategies: The first strategy is ›visual awe‹ (usage of drone shots, time lapses, b-roll, professional shots) and needs a diversified and more expensive production apparatus as well as great scenery (Sailing la Vagabond, Casey Neistat, Marques Brownlee). The second strategy is ›excitement‹, which refers to the delivery of a strong theatrical performance and/or funny nonsense and can be realized with both amateurish and professional visuals (Markiplier, Elle Mills, Kelly Stamps). The third strategy is the ›conveying of information‹, which can be delivered in both a calm and animated manner. Visually, it tends to be plain, simple, and authentic (Anna's Analysis, Tara Brabazon). All these strategies overlap and co-exist.

Across these major visual strategies, two different styles could be identified: First, the vlogs tend to be visual mainstream and ›hygge style‹. They do so to avoid potentially alienating viewers. The hygge aesthetic is calm, relaxed, and pastel. Hygge represents a string of popular culture which is neither rooted in subversive punk nor in bourgeois author style cinema. It involves hand-lettering pastel typography, which is to contribute to a self-aware authenticity.

Second, and in contrast, the ›attention-grabbing style‹ features high-contrast coloring and action-rich imagery. It doesn't even try to be subtle and delivers an intense and diverse color palette. Bold, non-serif typography is optimized for visibility in video preview titles. It can include anything that is considered visually extreme, such as an exalted, screaming expression.

Vlog aesthetics are also informed by a myriad of storytelling formats. Monologue, Q+A, confession, A day in life, reactions, rants, reportage, let's play, comedic sketches and pranks provide a broad range of narrative devices. The richness in storytelling can appear as eclectic and random. It is up to the creators/protagonists and their individual subjectivity to unify and synthesize the diverse narrative approaches. Therefore, it can be argued that successful vloggers need a broad ability to integrate differences – visual, thematic or narrative. From the reviewed material it can be concluded that ›theatrical performance‹ is at the center of this ability. Although ›cinematic‹ is publicly celebrated in vlogging ›discourse‹, many creators also build on theatrical rather than cinematic techniques to convey emotions and cause affective interpellation. A truly cinematic approach, which would convey emotions through montage, would be too expensive for many YouTubers. Emotions thus get spoken directly to the cam. One of the most frequent exclamations of emotion in Sailing La Vagabonde is »Awesome!«. To »conceal from our audience all evidence of ›dirty work‹« (Goffman 1956, 28) is a prerequisite for YouTubers. Cutting and editing footage makes it possible to compress the viewer's experience, omit the tedious and boring tasks that led to its completion and discard insecurities and mishaps.

### 3 Episodes

The future scenario of *Data Proxy* is not in itself dystopian. It is rather an extrapolation based on today's developments. It takes the slogan ›Data is the new oil‹ seriously and considers data as a productive means and a productive force.<sup>44</sup> When data is the new oil, what about oil spills, fracking, pollution, worker safety and oil prices? *Data Proxy* is less interested in surveillance, since this is already a widely-researched field.<sup>45</sup> Rather, it looks into a situation where the production of data has become a necessity both for individuals and society. Why would such a scenario not be entirely dystopian? While dystopian surveillance scenarios may serve as a warning, they also tend to lock the future. Instead of attempting to identify an evil and powerful agent (as in conspiracy theories), the *Data Proxy* scenario investigates the loopholes, glitches, and breakdowns encountered in an imaginary scenario where datafication is all-encompassing. Tactics for dealing with datafication are demonstrated in today's society, for example, by the gig workers organizing with each other using their employers' apps through off-label use, or outsmarting the Uber app (Scholz 2016), or by high school students tricking automated graduation software by delivering convenient keywords (Chin 2020). Scholars have identified a tendency to perpetuate normativity in algorithmic figurations in relation to gender, race and other social normative constructions (Noble 2018). *Data Proxy* aims to identify and exemplify other scenarios in order to strengthen and re-amplify these positions of resistance.

#### 3.1 Episode Development Notes

The following section has been kept rather short. It describes character and script development, production, editing and post-production and includes the full scripts for the six *Data Proxy* episodes.

##### 3.1.1 Character and script development

A range of decisions have been made: The main character speaks from the perspective of a partly precarious, self-employed freelancer but also that of a white, privileged, middle-class CIS male. He is into the latest tech and trends, which befits his occupation as tech entrepreneur. It was also clear early on that an ›artsy‹ visual approach was to be used and the commercial influencer fully submerged in visual language and content. Therefore, in turn, the character and script development needed to be adjusted for a consistent, continuous narration. This intentional figuration resonates with the artistic strategies outlined in chapter 1.2.2 *Artistic continuities: overidentification and narration*.

The *Data Proxy* is looking for a unicorn client who will make him rich. It looks as if the CEO of a dating website could fit the bill. While in the beginning the main character is self-centered and focused on his own success, the introduction of the supporting character Margret from the ›Data Equality

---

<sup>44</sup> The slogan itself is a false metaphor, as discussed in *Epistemic Harvest – The electronic database as discourse and means of data production* (Hunger 2018).

<sup>45</sup> In the US, the research field of ›surveillance studies‹ has emerged in social sciences. In my opinion, it explains away the structural inequality inherent to capitalism and capital by re-focusing on surveillance. For a critique, see for instance Morozov's review of Zuboff's recent book (Morozov 2019).

Foundation« offers an additional perspective, which includes resistant solidarity and modes of self-organization. The main character faces the dilemma of choosing between his own aspirations for success and the needs of others. He has to determine how much care and free labor he will provide, and if he wants to continue adhering to the data production principles, despite all the consequences. The *Data Proxy* is drawn to Margret the NGO founder on a personal level, which adds drama and results in a reformulation of his needs, with a shift away from the initial dream of wealth and prosperity. This development seems partly contradictory, but these contradictions aim to make the figure complex and multi-dimensional. The decisions made for the character resonate with Memo Atken's idea of ›data dramatization« as outlined above in chapter 1.3 »*What that means to you*« – *Towards the dramatization of data practices*

Aspects of the fictive data production scenario are developed within this framework. The work aims to stimulate discussion about the scenarios, leaving the viewer in a state of suspense where s/he can't decide whether *Data Proxy* is fiction or reality. A suspended state of ›truth« is at the center of *Data Proxy*'s fictional world. Since vlogging is also always entertainment, episodes aim for speed, some humor and the feel of vlogs, which is not only related to the scenario, but also to delivery.

There have been specific limitations to character development. For budgetary reasons and out of artistic curiosity, it was clear from the beginning that I would perform the *Data Proxy* myself. My appearance necessarily deviates from the normative of influencers as young and good-looking. Working with an actor could have led to greater plausibility but this was impossible for a no-budget production. In order to create a character and not simply double myself, I invested in clothing and a personal expression that would show a different me. Therefore, I researched current outfits and styles, mostly in red and brownish colors, and most notably invested in a specific hat, an influencer hat, which you wouldn't normally see on the streets of Leipzig (Fig. 19 and Fig. 20). For each episode, there is a slight variation in the outfit.

The character is also presented through a series of photographs which were used for the website [www.dataproxy.biz](http://www.dataproxy.biz) and the vlog intro trailer, shot by the professional photo artist Walther Le Kon. These include the *Data Proxy* posing in urban scenes such as a baseball field or next to graffiti and carrying around youthful accessories, such as a longboard. A scene next to a Starbucks demonstrates the ability to buy overpriced coffee – a sign of success and hard work.

For the fictional world, the character had to meet other people, an aspect which I underestimated during the initial development, and which only emerged after the first editing and test episodes. As such, another revision of the script involved more external characters who would create interactions with the main character and thus drive character development.

The script developed through iterative steps following the previously established general outline. For each episode one major topic was identified, researched as necessary (i.e., I had to find out if data toilets actually already existed and it turned out that the concept was already in circulation), and written. Subsequently, for each episode the framing (a place or an action) and the general plot which evolves from episode to episode included around the framing, were determined. The writing process also included decisions about incorporating formats typical of vlogs, such as Ask Me Anything (which was discarded in the editing process), drone flights, or guest interviews. These scripts were discussed with colleagues and mentors and further revised.

In total the material could have easily filled eight to nine episodes, and shortening and streamlining the episodes required significant effort.





Fig. 19: The *Data Proxy* outfit, with an orange-brimmed hat and expensive coffee (Photo: Walther Le Kon, with permission).



Fig. 20: Another variation of the outfit. (Photo: Walther Le Kon, with permission).

### 3.1.2 Production

The individual episodes have been developed in a longer process, further delayed by limitations of the Covid pandemic. With a home office and home schooling it became difficult to create space for artistic creativity and also for simple organizational tasks. Cinematic sets, such as a café or an office space, became temporarily unavailable or required special arrangements. The initial decision was to make the production as pared-down as possible, which resulted in a one-person effort. Such an approach reflects the realities of many vloggers, who in their initial efforts work mostly alone and don't hire a team until they have experienced reasonable growth. In hindsight, it was a fortunate decision because the independent DIY approach made it possible to continue working through the pandemic without the additional responsibilities for a team which would have had necessarily resulted.

After creating the episode outlines, which underwent three or more iterations, and writing the dialogue and monologue in a screenplay script, a list of possible scenes and scenery was identified. Step-by-step episodes were recorded, which involved micro-logistics such as transport, wardrobe, teleprompter preparation, camera and sound setup and the actual recording. Recordings had to be repeated because the camera had no autofocus in movie mode and also due to consistent sound problems. When interview partners or co-actors were involved, additional communication and recording became necessary, increasing the overall complexity.

Filming was done with an older Canon D5 Mark II DSLR camera with a 30mm lens which was available, though it was not the best possible equipment. For instance, that camera did not have autofocus in video mode, so more often than not the material was out of focus. The camera also had a fixed monitor on the back, while today's vlogging cameras feature a swiveling monitor so that a performer can actually see him/herself performing. In addition, a directed microphone was used, as well as a tripod. While a tripod creates a rather static, fixed, and cinematographic look, it was countered by using the camera in a more dynamic mode, holding it in one hand. Material with a GoPro action cam was shot but not used in the final vlogs, since the wide-angle aesthetics constituted a departure from the other material. In hindsight, a lighter and more vlogging-oriented camera would have made a difference.

Production also involved aspects of performance. While some performances took place in safe, closed spaces at home or at a rented office space, others were in public, and as such became semi-public events. People passing by became interested and watched the performance. I do not considering myself a professional performer, so these situations resulted in a lot of personal pressure and became rather stressful. Another aspect of performance were repetitions. For each scene, about three to five different shots were performed from which the best shot could be selected during the editing process. Here, vlogging clearly distinguishes itself from a purely documentary approach, where such a staging would be avoided. As the text was performed, the wording was partially adapted or edited for the teleprompter or else the decision was made to deliver specific parts without the teleprompter. Following in the footsteps of a lot of vloggers (see previous chapter), the need for theatricality in a relatively fixed and limited image frame almost naturally led to the use of a lot of hand-gesturing and gesticulation during the performative parts. This aspect of delivery emerged rather spontaneously during the recording and was not pre-planned. Following early screen tests and evaluation, strong gesticulation was used more intentionally.

In some recordings the sound quality turned out to be lower than expected, with a lot of background noise or reverberation (in an office space), and this aspect could only be mitigated by re-recording one

part of an episode. Buying or renting further equipment was not possible and the sound was sometimes left as it was.

The process showed that single-person vlog production, when oriented towards the »cinematographic style« as discussed in section 2.3.2 *Cinematography oriented in a dynamic setup* is intensive and basically a full-time job. After this experience, it became far easier to appreciate the achievements of vloggers like Elle Mills, Sailing La Vagabonde and Casey Neistat and their contributions to vlogging as a visual culture.

### 3.1.3 Editing and Postproduction

Unlike daily or weekly vlogs, the editing process for *Data Proxy* was not linear. While producing vlogs on a daily basis, Casey Neistat edited (or later on worked with an editor) the material mostly the same day it was shot. Sailing La Vagabonde recorded and edited most of their material a few weeks in advance and in chronological order before putting it online. With its fixed script, *Data Proxy* differs from these approaches as episodes were filmed and edited in a non-chronological order and the six episodes were completely finished before being put online every two weeks. Most of the episodes were reviewed by colleagues and mentors before publication and revised slightly.

Editing was done with a MacBook Pro computer using Final Cut Pro and without any further specialized equipment. In the editing process, attempts were made to replicate the typical vlogging style using jump cuts, zoom ins, time-lapses, swish-pans and music-oriented cuts as observed in chapter 2.3.2.3 *Cinematographic qualities, including composition, scenery, montage, sound* and 2.3.3 *Conclusion*. As all the bits and pieces were coming together, editing and cutting became a rather light-hearted part of the process, which allowed for intuitive decisions and revisions. In parallel, post-production began with creating the visual appearance and screenshots for the YouTube platform, deciding on titles and determining the layout.

Post-production also required that the external music used be cleared and credited. Most of it was from YouTube's own library and some of it came from third-party creators. The music selected did very intentionally not follow my own personal preferences and instead mimicked the selection of famous YouTubers. For instance, one particular song which was used by Casey Neistat, a remix by the artist Dyalla of Frank Sinatra's *New York, New York*, was used. It clearly appropriates the sound of success.

Another aspect of postproduction was the development of the visual appearance. Since the narration emerges through a fictional aspiring business, a corporate identity was needed. The *Data Proxy*'s logo was developed as an anchor for a broader visual identity. It was important for the logo to have colors that contrasted with those in the vlogs, and for this reason a color gradient that transitions from a dark violet to a mint green was established (Fig. 22). These colors temper the emotional coloring of the *Data Proxy*'s clothes, which tend towards warmer hues such as red, orange and brown (Fig. 19 and Fig. 20), adding a professional look that is »cool«, relaxed and fresh.

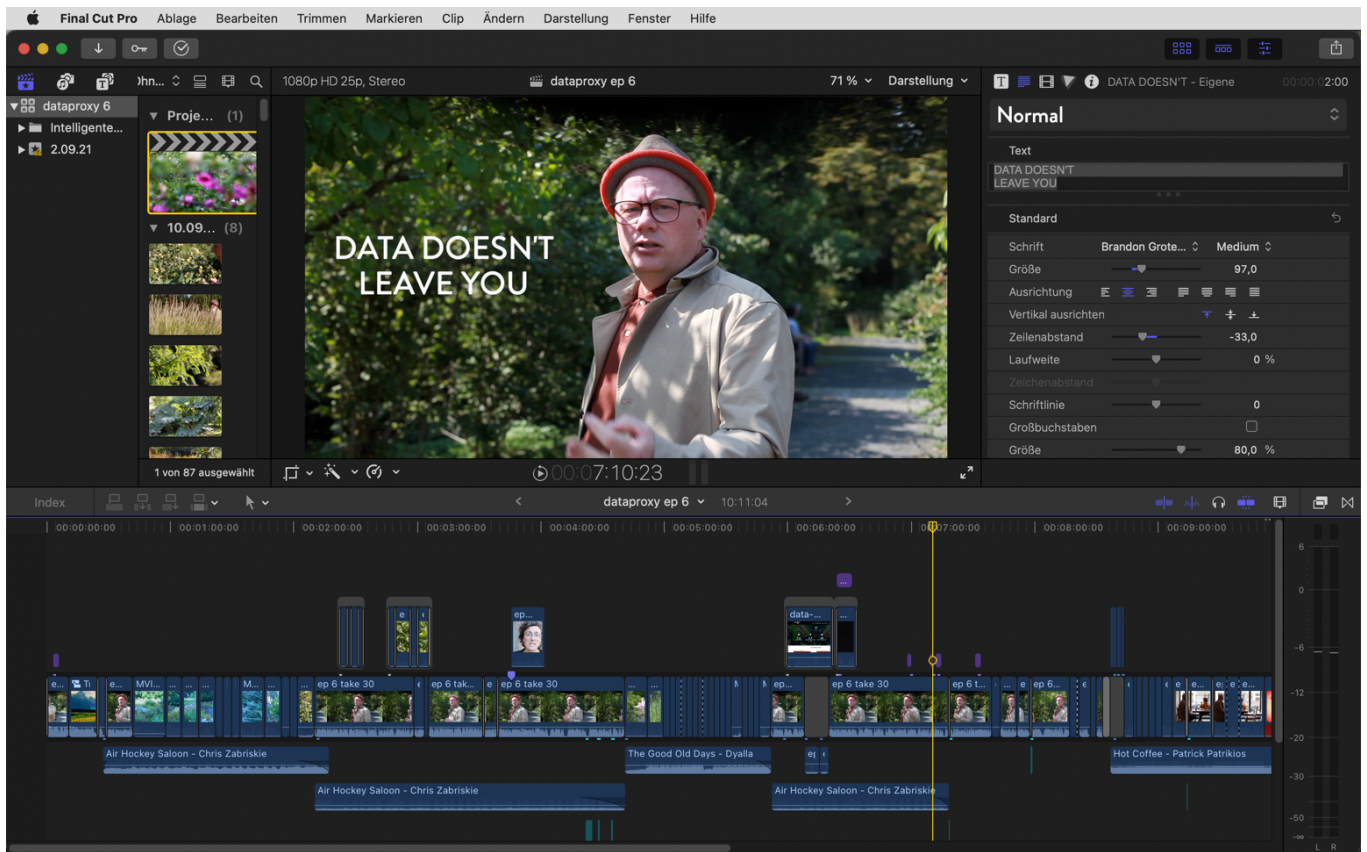


Fig. 21: Video editing software Final Cut Pro – User interface with timeline of Episode 6 (screenshot).

The Logo needed to work in smaller and in larger sizes against a range of photographic and cinematic backgrounds, so it was decided to use a bold font and to fill the counters of the letters with a color gradient as well. From there, the complete visual identity of the website was determined, concentrating on the Brandon Grotesque Medium font, which was then also used within the videos for titles. This is a geometric sans-serif font that draws on the modernist style of the Bauhaus era. It can be read clearly on different screen sizes and has distinctive features such as its slightly rounded-off corners, which smooth out the geometric constructiveness of the font.

Post-production also included the creation of a website at the URL <https://dataproxy.biz>. The author designed and programmed it by using Wordpress. The website looks like a normal business website and contains information about the *Data Proxy's* services (Fig. 22). It contains the sections ›Welcome‹, which is the home page that sets up the narration, ›Services‹, which offers nudging, unique data production, data coaching and emergency data production, ›Data Proxy Universe‹, which is where the vlogs get published, ›Contact‹ and ›Imprint / Credits / Privacy‹. It also shows exaggerated branding using fictional prizes won and fictional personal recommendations, for example. However, it never becomes too obvious that this could in fact be an art project.

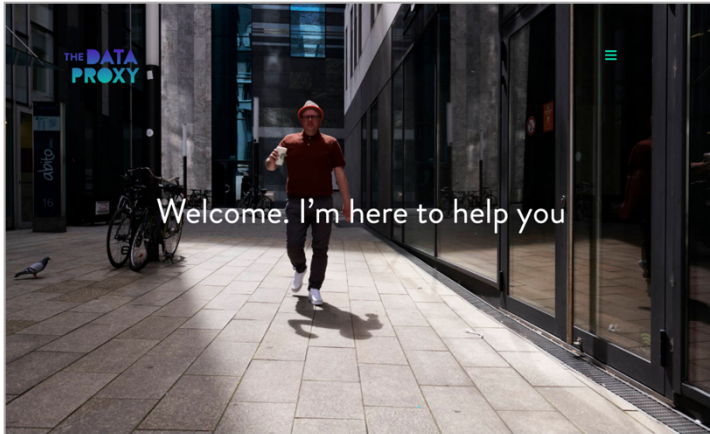


Fig. 22: Partial Screenshot of the data proxy website's homepage at <http://www.dataproxy.biz> with the lower end cropped (author, screenshot)

## Data Proxy

### Consultant & Vlogger

I help people to produce enough high quality data and to register it with the data authority. In doing so, I advise companies and individuals on their path to success.

My topics are data analysis and development as well as data cleansing.

I can help people to identify and realize their data potential.

I support companies in the selection of data strategies and advise individuals on their individual data hygiene.

I can clarify how aggregated data and meta-data shall be treated, how it is compensated, and when your data must be deleted.

Let's create greatness.



## Consulting to the extreme



The data production law assigns you to produce a certain amount of data each year. In addition to consulting on data production itself, I make sure your data is actually attributed to you. I can clarify the purposes your data may be used for.

## The Solution Scheme

How can I solve your problem?

The process diagram shows the elements that are crucial for your success to unfold.



- 1 **Analyze!**  
Understand potentials, recognize and correct errors!
- 2 **Create!**  
Generate high quality data and report it to the data authorities!
- 3 **Succeed!**  
Proactively outperform sloppy competitors!

The website builds strongly on principal photos of the *Data Proxy* character. It emerges from contemporary photographic language, developed in cooperation with the photographer Walter Le Kon, using urban activities such as skateboarding or having a coffee outdoors or urban backdrops such as graffiti. The leading question here was: How would a vlogger or influencer present himself, within the specific conditions which my body would allow, and which would potentially appeal to a broader audience? The website contains a section which embeds the six *Data Proxy* vlogs from YouTube which makes it possible to refer back and forth between the different data ecologies of the large-scale platform YouTube and the self-created and self-owned website.

Finally, post-production also included determining how the vlog would appear on YouTube. Building on the existing visual identity by using the same font and screenshots from the vlogs, the major contribution to visual identity within YouTube's visual ecology are the video preview pictures. These have been composed to show the logo, title and the episode number in front of a picture that shows the *Data Proxy* gesticulating theatrically. This is similar to the usage, for example, of icons in Anna's Analysis, as discussed in chapter 2.3.1.1 *Visual Language incl. Typography and Style*. The titles of the episodes have also been deliberately selected for punchiness and controversy, as was learned from Elle Mills and Casey Neistat. Episode 1, for instance, started with the name »Amazing Ways To Create Data« but eventually became »More Data!«. Other episodes are titled: »The Data Toilet«, »Data Poor?«, »Meeting a Professor«, »20K EuroCoin lost«, and »Data Socialist?«. The visual appearance within YouTube also asked for a profile picture (a headshot of the *Data Proxy*) and an end trailer in the video which would make it possible to place overlays that would lead to further content such as the subsequent episodes. The episode development notes call for a rather short conclusion since they mostly describe a production process which in one way or another resembles a typical vlog production pipeline, with the difference being that a visual concept was established from the beginning, whereas it usually forms over months or years of posting content.

### 3.2 Episodes Outline

<b>Episode</b>	<b>Story Arc</b>	<b>Data Arc</b>
1 – More Data!	<ul style="list-style-type: none"> <li>• Introduction: Explain what a <i>Data Proxy</i> is and does, where he lives, who he got educated and what are his character traits</li> </ul>	Data Creation: <ul style="list-style-type: none"> <li>• Logger Swing,</li> <li>• Data Black Market</li> </ul>
2 – The Data Toilet	<ul style="list-style-type: none"> <li>• <i>Data Proxy</i> learns about the Data Equality Foundation NGO</li> <li>• Meeting a fellow entrepreneur, Gabriel from Art Up Nation</li> <li>• Revealing that he has a new client.</li> </ul>	<ul style="list-style-type: none"> <li>• Data Doubles</li> <li>• Data Toilet</li> </ul>
3 – Data Poor?	<ul style="list-style-type: none"> <li>• Meeting with the Data Equality Foundation’s director Margret</li> </ul>	<ul style="list-style-type: none"> <li>• Data Company</li> <li>• Data distribution</li> </ul>
4 – Meeting a Professor	<ul style="list-style-type: none"> <li>• Short mentioning of client work</li> <li>• Rant: comfortable is lazy</li> </ul>	<ul style="list-style-type: none"> <li>• Interview with Professor Susan Calvin on social repercussions</li> </ul>
5 – 20,000 Eurocoin lost!	<ul style="list-style-type: none"> <li>• <i>Data Proxy</i> cancels the dating website job on learning that they betray their clients.</li> </ul>	<ul style="list-style-type: none"> <li>• Automation with Oracle Puppeteer</li> <li>• Nudging me – nudging you</li> </ul>
6 – Data Socialist?	<ul style="list-style-type: none"> <li>• Hanging out at the Botanical Garden</li> <li>• Him and Margret meet in a private setting</li> </ul>	<ul style="list-style-type: none"> <li>• Data ownership</li> </ul>

## 4 Summary

Building on the notion of data dramatization, *Data Proxy* is a narrative media archeological art project. It looks at the cultural forms, specifically at vlogs, that emerged and co-exist with data driven advertisement and entertainment platforms like YouTube. The figure of the *Data Proxy* is a stand in for a larger social problem of the near future – a total datafication of life.

Different from a purely sociological analysis this artistic research project intervenes into the researched field. It adds subtle provocations and stays just an inch from activism. By using an unrivaled online platform, YouTube, *Data Proxy* aims to reach a broader audience than just an art venue or an scientific audience. This decision leads to the consequence of having to adapt to the platforms visual and data ecology.

The documentation described a series of Vlog aesthetics and creative strategies used by Vloggers, which adapt to YouTube's platform ecology. The following vlogs were reviewed: Anna's Analysis (veganism, van life, soul searching), Marques Brownlee (tech), Elle Mills (pranks, LGBTQ, coming-of-age), Casey Neistat (travel, entrepreneur), Kelly Stamps (fashion, minimalism), Sailing la Vagabonde (travel, family), Tara Brabazon (academic life) and Markiplier (gaming). The analysis differentiated between two vlog styles: 1.) Talk in a static setup and 2.) Cinematography oriented dynamic setup. While obviously both styles show intersections, they also differed in their particular visual language, typography and style, in how the actors performed their bodies and in cinematic qualities such as composition, scenery, montage and sound.

Their creative strategies are largely oriented on the platform economy and on their peers, so the innovation they brought are innovations in attention production which translates into the creation of transactional data using computational capital. The documentation described how creative strategies co-develop with the platform's attention economy and thus delivers an attempt to a political economy of creativity on online-platforms.

The Database as aesthetic media practice asks Vloggers for clearly identifiable personas and to aesthetic strategies of attention grabbing. These appear both through excited and calm delivery. In addition to Uber drivers and amazon workers, Vloggers are examples of working under constant pressure to be evaluated by data metrics. Through their creative involvement they are investing not just labor in general, but specific creative, emotional and subjective labor. Vloggers and creators are a particularly obvious example for a data ecology through which subjectivity is captured into data to be subsequently transformed into value and profit.

This artistic research project concludes that the media impact and the platform-specific creativity of Vlogging deserves more research. It is possible to mimic the performance aspect of Vlogging while the performative part like age, gender and social conventions bounds the creators to their bodies and habitus. Intervening with the figure of the *Data Proxy* was an intense artistic endeavour, since different to text based research the artist exerted the power and underwent the violence of performing for a data-driven platform.



## Bibliography

- Akel, Joseph. »Taking Stock with DIS.« *Interview Magazine*, 25 Feb. 2013, <https://www.interviewmagazine.com/art/dis-suzanne-geiss>.
- Akten, Memo. »Data Dramatization.« *Medium*, 18 Oct. 2015, <https://medium.com/@memoakten/data-dramatization-fe04a57530e4>.
- Alexander, Julia. »YouTube's Top Creators Are Burning out and Breaking down En Masse.« *Polygon.Com*, 1 June 2018, <https://www.polygon.com/2018/6/1/17413542/burnout-mental-health-awareness-youtube-elle-mills-el-rubius-bobby-burns-pewdiepie>.
- Alphabet Inc. *Alphabet Inc – Annual Report Pursuant to Section 13 or 15(d) of the Securities Exchange Act of 1934 For the Fiscal Year Ended December 31, 2019*. Feb. 2020, [https://abc.xyz/investor/static/pdf/20200204\\_alphabet\\_10K.pdf](https://abc.xyz/investor/static/pdf/20200204_alphabet_10K.pdf).
- Andreas, Michael. »Flächen/Rastern Zur Bildlichkeit Der Drohne.« *Universitätsbibliothek Freiburg*, 2015, <https://doi.org/10.6094/BEHEMOTH.2015.8.2.872>.
- Apprich, Clemens. »Remaking Media Practices – From Tactical Media to Post-Media.« *Mute*, 14 Feb. 2013, <https://www.metamute.org/editorial/lab/remaking-media-practices-%E2%80%93-tactical-media-to-post-media>.
- Arlander, Anette. *Artistic Research and/as Interdisciplinarity*. Faculty of Fine Arts University of Porto, 2016.
- Arns, Inke. *Netzkulturen*. Europäische Verlagsanstalt, 2002.
- . *Objects in the Mirror May Be Closer than They Appear*. Humboldt-Universität zu Berlin, Philosophische Fakultät II, 22 Nov. 2004, <https://doi.org/10.18452/15154>.
- Arns, Inke, and Irwin, editors. *Irwin Retroprincip 1983 - 2003*. Revolver, Archiv für Aktuelle Kunst, 2003.
- Bacon, Jane, and Vida L. Midgelow. *Reconsidering Research and Supervision as Creative Embodied Practice – Reflections from the Field*. Artistic Doctorates in Europe, 2019, [www.artisticdoctorates.com](http://www.artisticdoctorates.com).
- Berryman, Rachel, and Misha Kavka. »Crying on YouTube – Vlogs, Self-Exposure and the Productivity of Negative Affect.« *Convergence: The International Journal of Research into New Media Technologies*, vol. 24, no. 1, Feb. 2018, pp. 85–98, <https://doi.org/10.1177/1354856517736981>.
- Bevan, Jennifer L. »Perceptions of Selfie Takers versus Selfie Stick Users – Exploring Personality and Social Attraction Differences.« *Computers in Human Behavior*, vol. 75, Oct. 2017, pp. 494–500, <https://doi.org/10.1016/j.chb.2017.05.039>.
- Bishop, Claire. »Digital Divide.« *Artforum*, Sept. 2012, pp. 434–42.
- Blu Toys Club Surprise. *Surprise Play Doh Eggs Peppa Pig Stamper Cars Pocoyo Minecraft Smurfs Kinder Play Doh Sparkle Brilho*. 2016, <https://www.youtube.com/watch?v=IYjxnvH0dYE>.
- Bolt, Barbara. *A Performative Paradigm for the Creative Arts?* Working Papers in Art and Design. No 5, 2008, <https://theearoftheduck.files.wordpress.com/2012/10/a-performative-paradigm-for-the-creative-arts.pdf>.
- Boyd, Danah, and Kate Crawford. »Critical Questions for Big Data – Provocations for a Cultural, Technological, and Scholarly Phenomenon.« *Information, Communication & Society*, vol. 15, no. 5, June 2012, pp. 662–79, <https://doi.org/10.1080/1369118X.2012.678878>.
- Brabazon, Tara. *Vlog 228 - Productivity during Tough Times*. 2020, <https://www.youtube.com/watch?v=UY8AgUK7r7E>.
- Branigan, Edward. *Point of View in the Cinema – A Theory of Narration and Subjectivity in Classical Film*. Mouton, 1984.
- Brecht, Bertolt. »Über Stoffe Und Form.« *Werke*, edited by Werner Hecht, vol. Bd. 21: Schriften I, Aufbau/Suhrkamp, 1992, pp. 302–04.
- Broeckmann, Andreas. *Postmedia Discourses – A Working Paper*. 7 July 2013, [http://abroeck.in-berlin.de/wp-content/uploads/2020/11/Broeckmann\\_Postmedia\\_Discourses\\_v2\\_2013.pdf](http://abroeck.in-berlin.de/wp-content/uploads/2020/11/Broeckmann_Postmedia_Discourses_v2_2013.pdf).
- Brownlee, Marques. *On the iPhone 12 With No Charger*. 2020, <https://www.youtube.com/watch?v=8IB7JcKJEeI>.
- Bucher, Taina. »Objects of Intense Feeling – The Case of the Twitter API.« *Computational Culture*, 2013, <http://computational-culture.net/article/objects-of-intense-feeling-the-case-of-the-twitter-api>.
- Burgess, Jean, and Joshua Green. »The Entrepreneurial Vlogger – Participatory Culture Beyond the Professional-Amateur Divide.« *The YouTube Reader*, edited by Pelle Snickars and Patrick Vonderau, National Library of Sweden, 2009, pp. 89–108.
- Butler, Judith. *Das Unbehagen Der Geschlechter*. Translated by Kathrina Menke, Suhrkamp, 1991.

- Carasu, Elayna, and Riley Whitelum. *Calm Before the Storm (Sailing La Vagabonde) Ep. 100*. 2017, <https://www.youtube.com/watch?v=olOYVEcF1M0>.
- . *CLOSE CALL with a TIGER SHARK in the Bahamas!!!* 2018, <https://www.youtube.com/watch?v=H2Tx4O1nFdU>.
- . *Dodging STORMS with Greta Thunberg // 60 Knot Winds & 7m Waves on the Horizon!! Ep.2*. 2020. YouTube, [https://www.youtube.com/watch?v=TUte8inU\\_DA](https://www.youtube.com/watch?v=TUte8inU_DA).
- . *What the F#\*% Do We Do? [Ep 136]*. 2018, <https://www.youtube.com/watch?v=ibYMzXuS0GQ>.
- Cazeaux, Clive. »Inherently Interdisciplinary – Four Perspectives on Practice-Based Research.« *Journal of Visual Art Practice*, vol. 7, no. 2, Nov. 2008, pp. 107–32, [https://doi.org/10.1386/jvap.7.2.107\\_1](https://doi.org/10.1386/jvap.7.2.107_1).
- Chaslot, Guillaume. »How YouTube's AI Boosts AlternativeFacts.« *Medium*, 3 Apr. 2017, <https://medium.com/@guillaumechaslot/how-youtubes-a-i-boosts-alternative-facts-3cc276f47cf7>.
- Chin, Monica. »These Students Figured out Their Tests Were Graded by AI — And the Easy Way to Cheat.« *The Verge*, 2 Sept. 2020, <https://www.theverge.com/2020/9/2/21419012/edgenuity-online-class-ai-grading-keyword-mashing-students-school-cheating-algorithm-glitch>.
- Cochior, Cristina, and Ruben van den Ven. »Introduction – Plotting Data.« *Plotting Data*, 2019, <http://plottingd.at/a/introduction.html#the-dramaset>.
- Couldry, Nick, and Ulises Ali Mejias. *The Costs of Connection - How Data Is Colonizing Human Life and Appropriating It for Capitalism*. Stanford University Press, 2019.
- Covington, Paul, et al. »Deep Neural Networks for YouTube Recommendations.« *Proceedings of the 10th ACM Conference on Recommender Systems - RecSys '16*, ACM Press, 2016, pp. 191–98, <https://doi.org/10.1145/2959100.2959190>.
- Daniels, Dieter, et al. »Media Art Net | Concept.« *Media Art Net*, 2005 2001, <http://www.medienkunstnetz.de/concept>.
- Daub, Adrian. *What Tech Calls Thinking*. FSG Originals, Farrar, Straus and Giroux, 2020.
- Duguay, Stefanie. »You Can't Use This App for That – Exploring Off-Label Use through an Investigation of Tinder.« *The Information Society*, vol. 36, no. 1, Jan. 2020, pp. 30–42, <https://doi.org/10.1080/01972243.2019.1685036>.
- Eckel, Julia, et al., editors. *Exploring the Selfie*. Springer International Publishing, 2018.
- Etzemüller, Thomas. »It's the Performance, Stupid« Performanz → Evidenz.« *Der Auftritt*, edited by Thomas Etzemüller, transcript Verlag, 2019, pp. 9–44.
- Facebook for Business. »About Facebook Pixel.« *Facebook Business Help Centre*, <https://en-gb.facebook.com/business/help/742478679120153>. Accessed 6 June 2019.
- Fisher, Eran. »Audience Labour on Social Media – Learning from Sponsored Stories.« *Reconsidering Value and Labour in the Digital Age*, edited by Eran Fisher and Christian Fuchs, Palgrave Macmillan, 2015, pp. 115–32.
- Fuchs, Christian. »The Digital Labour Theory of Value and Karl Marx in the Age of Facebook, YouTube, Twitter, and Weibo.« *Reconsidering Value and Labour in the Digital Age*, edited by Eran Fisher and Christian Fuchs, Palgrave Macmillan, 2015.
- Gerlitz, Carolin, and Anne Helmond. »The Like Economy – Social Buttons and the Data-Intensive Web.« *New Media & Society*, vol. 15, no. 8, Dec. 2013, pp. 1348–65, <https://doi.org/10.1177/1461444812472322>.
- Gitelman, Lisa. *Raw Data Is an Oxymoron*. MIT Press, 2013. Bücherregal.
- Goffman, Erving. *The Presentation of Self in Everyday Life*. University of Edinburgh/Social Sciences Research Centre, 1956.
- Google Inc. »AdSense Revenue Share.« *AdSense Help*, <https://support.google.com/adsense/answer/180195?hl=en>. Accessed 1 Oct. 2020.
- Goriunova, Olga. »The Digital Subject – People as Data as Persons.« *Theory, Culture & Society*, vol. 36, no. 6, Nov. 2019, pp. 125–45, <https://doi.org/10.1177/0263276419840409>.
- Gray, Colin M., et al. »The Dark (Patterns) Side of UX Design.« *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*, ACM Press, 2018, pp. 1–14, <https://doi.org/10.1145/3173574.3174108>.
- Gregg, Melissa, and Gregory J. Seigworth, editors. *The Affect Theory Reader*. Duke University Press, 2010.
- Gregory, Derek. »Drone Geographies.« *Radical Philosophy*, no. 183, Feb. 2014, pp. 7–19.

Haraway, Donna. »Situated Knowledges – The Science Question in Feminism and the Privilege of Partial Perspective.« *Feminist Studies*, vol. 14, no. 3, Autumn 1988, <https://doi.org/10/bvtwq4>.

Harger, Honor. »Unmanned Aerial Ecologies.« *Honor Harger*, 16 Jan. 2014, <https://honorharger.wordpress.com/2014/01/16/unmanned-aerial-ecologies-2/>.

Hoff, Todd. »YouTube Architecture.« *High Scalability*, 12 Mar. 2008, <http://highscalability.com/youtube-architecture>.

Holt, Jennifer, and Patrick Vonderau. »»Where the Internet Lives« – Data Centers as Cloud Infrastructure.« *Signal Traffic – Critical Studies of Media Infrastructures.*, edited by Lisa Parks and Nicole Starosielski, University of Illinois Press, 2015.

Hunger, Francis. »Deep Love Algorithm Exhibition and Database Derive in Berlin.« *Databasecultures*, 7 Feb. 2015, <http://databasecultures.irmielin.org/deep-love-algorithm-in-berlin/>.

---. »Epistemic Harvest – The Electronic Database as Discourse and Means of Data Production.« *A Peer Reviewed Journal About Research Values*, vol. 7, no. 1, July 2018, <https://doi.org/10.7146/aprja.v7i1.115064>.

Jacobs, Jerry A., and Scott Frickel. »Interdisciplinarity – A Critical Assessment.« *Annual Review of Sociology*, vol. 35, no. 1, Aug. 2009, pp. 43–65, <https://doi.org/10/c8dkjs>.

Jones, Jonathan. »'I've Seen More Self-Aware Ants!' AI: More Than Human – Review.« *The Guardian*, 15 May 2019. *The Guardian*, <https://www.theguardian.com/artanddesign/2019/may/15/ai-more-than-human-review-barbican-artificial-intelligence>.

Kabakov, Il'ja Iosifovič, and Boris Grojs, editors. *Ilya Kabakov*. Phaidon, 1998.

Kurpiel, Ryszard. »The Stylistics of Selected American, Italian and Polish Challenge Vlogs.« *Styles of Communication*, vol. 9, no. 2, 2017, pp. 65–79.

Lange, Patricia G. »Algorhythmia.« *Video Vortex Reader III: Inside the YouTube Decade*, edited by Geert Lovink and Andreas Treske, Hogeschool van Amsterdam, Lectoraat Netwerkcultuur, 2020, pp. 50–59.

Laperdrix, Pierre, et al. *Beauty and the Beast – Diverting Modern Web Browsers to Build Unique Browser Fingerprints*. IEEE, 2016, pp. 878–94, <https://doi.org/10.1109/SP.2016.57>.

Lilienthal, Matthias, and Claus Philipp, editors. *Schlingensiefs Ausländer Raus – Bitte Liebt Österreich*. Suhrkamp, 2000.

Lovink, Geert. »Sad by Design.« *Eurozine*, 10 Jan. 2019, <https://www.eurozine.com/sad-by-design/>.

Luhmann, Niklas. »Konstruktion von Realität – Medien Als Gedächtnis von Gesellschaft.« *Die Realität der Massenmedien*, Opladen, Westdeutscher Verlag, 1996.

Lupton, Deborah. »Digital Companion Species and Eating Data – Implications for Theorising Digital Data–Human Assemblages.« *Big Data & Society*, vol. 3, no. 1, Jan. 2016, <https://doi.org/10.1177/2053951715619947>.

Markplier. *EVERYTHING IS FINE | Minecraft - Part 17*. 2020, <https://www.youtube.com/watch?v=0pt1nbi-GPI>.

---. *LIXIAN IS GONE LUNKY IS HERE | Minecraft - Part 18*. 2020, <https://www.youtube.com/watch?v=Yrd6JaTgN4w>.

---. *Someone Wrote a Book About Me... WITHOUT MY PERMISSION?!* 2020, <https://www.youtube.com/watch?v=5SP9ozIIrn0>.

Marwick, Alice Emily. *Status Update – Celebrity, Publicity, and Branding in the Social Media Age*. Yale University Press, 2013.

Mathur, Arunesh, et al. »Dark Patterns at Scale – Findings from a Crawl of 11K Shopping Websites.« *Proceedings of the ACM on Human-Computer Interaction*, vol. 3, no. CSCW, Nov. 2019, pp. 1–32. DOI.org (Crossref), <https://doi.org/10.1145/3359183>.

Mattes, Eva, and Franco Mattes. »Nike Ground (2003).« *0100101110101101.Org*, 2003, <https://0100101110101101.org/nike-ground>.

McHugh, Gene. *Post Internet – Notes on the Internet and Art*. Link Editions, 2011.

Mills, Elle. *Burnt Out At 19*. 2018, <https://www.youtube.com/watch?v=WKKwgq9LRgA>.

---. *COMING OUT (ELLE MILLS STYLE)*. 2017, <https://www.youtube.com/watch?v=pDi7cezk9CA>.

---. *I Learned a Foreign Language in 7 Days (Tagalog)*. 2020, <https://www.youtube.com/watch?v=sqqZfD0Hti0>.

---. *Meeting My Brother's New Girlfriend for the First Time*. 2019, <https://www.youtube.com/watch?v=JDWDYIBkIbw>.

---. *My First Kiss*. 2018, <https://www.youtube.com/watch?v=IFNvAhKHSTk>.

Morozov, Evgeny. »Capitalism's New Clothes – Shoshana Zuboff's New Book on Surveillance Capitalism Emphasizes the Former at the Expense of the Latter.« *The Baffler*, 2 Apr. 2019, <https://thebaffler.com/latest/capitalisms-new-clothes-morozov>.

Müller, Heiner. *Dichter Als Metaphernschleuder*. Interview by Alexander Kluge, 16 June 1997, <https://kluge.library.cornell.edu/conversations/mueller/film/101/transcript>.

Mulvey, Laura. »Visual Pleasure and Narrative Cinema.« *Screen*, vol. 16, no. 3, Sept. 1975, pp. 6–18, <https://doi.org/10.1093/screen/16.3.6>.

Musser, John, and Tim O'Reilly. *Web 2.0 – Principles and Best Practices*. O'Reilly Media, 2007.

Neistat, Casey. *HOW TO VLOG LIKE CASEY NEISTAT* by CASEY NEISTAT. 2018, <https://www.youtube.com/watch?v=Q980C74SdYQ>.

---. *Nintendo Switch on an Airplane*. 2017, <https://www.youtube.com/watch?v=9fjvwaNANxQ>.

---. *THE ONLY THING BETTER THAN A BOOSTED BOARD*. 2016, <https://www.youtube.com/watch?v=V2IbfugMGP4>.

Neural Editors. »Adversarial.Io, Subverting Image Recognition.« *Neural.It*, 1 Apr. 2021, <http://neural.it/2021/01/adversarial-io-subverting-image-recognition/>.

Nixon, Brice. »The Exploitation of Audience Labour – A Missing Perspective on Communication and Capital in the Digital Era.« *Reconsidering Value and Labour in the Digital Age*, edited by Eran Fisher and Christian Fuchs, Palgrave Macmillan, 2015.

Noble, Safiya Umoja. *Algorithms of Oppression – How Search Engines Reinforce Racism*. New York University Press, 2018.

Ohr, Roberto. *Phantom Avantgarde – Eine Geschichte Der Situationistischen Internationale Und Der Modernen Kunst*. Edition Nautilus, Galerie van de Loo, 1990.

Olson, Marisa. »Postinternet – Art after the Internet.« *Foam Magazine*, no. 20, Winter 2011, pp. 59–63.

Parikka, Jussi. *What Is Media Archaeology?* Polity Press, 2012.

Parks, Lisa. *Rethinking Media Coverage – Vertical Mediation and the War on Terror*. Routledge, Taylor & Francis Group, 2018.

Parks, Lisa, and Caren Kaplan, editors. *Life in the Age of Drone Warfare*. Duke University Press, 2017.

Paul, Christiane. *Digital Art*. Translated by Christian Werner et al., Deutsche Erstausgabe, Deutscher Kunstverlag, 2011.

Paul, Logan. *ALEX JONES IS ALEX JONES - IMPAULSIVE EP 60*. 2019, <https://www.youtube.com/watch?v=HmStbc0tNcA>.

Phelps-Ward, Robin J., and Crystal T. Laura. »Talking Back in Cyberspace – Self-Love, Hair Care, and Counter Narratives in Black Adolescent Girls' YouTube Vlogs.« *Gender and Education*, vol. 28, no. 6, Sept. 2016, pp. 807–20, <https://doi.org/10.1080/09540253.2016.1221888>.

Pietruszka, Magdalena. »Watching People Playing Games – A Survey of Presentational Techniques in Most Popular Game-Vlogs.« *Res Rhetorica*, edited by Agnieszka Kampka and Ewa Modrzejewska, Apr. 2016, pp. 54–69, <https://doi.org/10.17380/rr.2016.4.4>.

Pihlaja, Stephen. »Drama in the YouTube Community.« *Antagonism on YouTube – Metaphor in Online Discourse*, Bloomsbury Academic, 2014, pp. 1–14.

Polanyi, Michael. *Personal Knowledge – Towards a Post-Critical Philosophy*. Univ. of Chicago Press, 1958.

Poster, Mark. *The Second Media Age*. Polity Press, 1995.

Quaranta, Domenico. *Beyond New Media Art*. Link Editions, 2013.

Reid, Anna. *4 Ways Influencers Manipulate You*. 2020, <https://www.youtube.com/watch?v=cR2njp5uCvE>.

---. *Fake Positivity on Social Media*. 2019. *YouTube*, <https://www.youtube.com/watch?v=KqHskWgm0Gg>.

---. *I Joined The Breakaway Movement So You Don't Have To*. 2020, <https://www.youtube.com/watch?v=JTU0SC5SfQU>.

---. *The Reality of Van Life on Social Media*. 2019, <https://www.youtube.com/watch?v=KvOHjIGi30Y>.

Rieder, Bernhard, et al. »Mapping YouTube.« *First Monday*, July 2020, <https://doi.org/10.5210/fm.v25i8.10667>.

Rogers, Andrew, and Gary Brewer. »Google Analytics Usage Statistics.« *Built With*, 2018, <https://trends.builtwith.com/analytics/Google-Analytics>.

Rohde, Paul, and Gunnar Mau. »Wolves in Sheep's Clothing? Influencer Marketing beyond Endorsements – Deconstructing Persuasive and Questionable Marketing Strategies in Vlogs on YouTube Based on Social Influence Heuristics Framework.« *SSRN Electronic Journal*, 2019, <https://doi.org/10.2139/ssrn.3464820>.

- Rossiter, Ned. *Software, Infrastructure, Labor – A Media Theory of Logistical Nightmares*. Routledge Taylor & Francis Group, 2016.
- Schäfer, Fabian, and Peter Mühleder. »Konnektiver Zynismus Und Neue Rechte – Das Beispiel Des YouTubers Adlersson.« *Zeitschrift Für Medienwissenschaft*, vol. 12, no. 22–1, Jan. 2020, pp. 130–49, <https://doi.org/10.14361/zfmw-2020-120114>.
- Schelenz, Laura. »Consumerism, Sociability, and Techno-Optimism? An Ethical Analysis of the Promotion of Social Norms in YouTuber Zoella's Vlogs.« *Ethik in Serie – Eine Festschrift Zu Ehren von Uta Müller.*, edited by Cordula Brand and Simon Meisch, Universität Tübingen, 2018.
- Schmidt-Burkhardt, Astrid. *Die Kunst Der Diagrammatik. Perspektiven Eines Neuen Bildwissenschaftlichen Paradigmas*. Transcript, 2012.
- Scholl, Dennis. *Inside My Studio – Tom Sachs*. 2019, <https://www.youtube.com/watch?v=td7LrVY74Gg>.
- Scholz, Trebor. *Overworked and Underpaid – How Workers Are Disrupting the Digital Economy*. Polity Press, 2016.
- Schüttpelz, Erhard. »Die Medienanthropologische Kehre Der Kulturtechniken.« *Archiv Für Mediengeschichte. Themenschwerpunkt – Kulturgeschichte Als Mediengeschichte (Oder Vice Versa?)*, edited by Lorenz Engell et. al., Universitätsverlag Weimar, 2006, pp. 1–23.
- . »Infrastrukturelle Medien Und Öffentliche Medien.« *Media in Action (Pre-Publication)*, no. 0, 2016, pp. 1–21.
- Senft, Theresa M. »Microcelebrity and the Branded Self.« *A Companion to New Media Dynamics*, edited by John Hartley et. al., Wiley-Blackwell, 2013, pp. 346–54.
- Shulgin, Alexei. »Art, Power and Communication.« *Ostranenie 97 – Das Internationale Forum Elektronischer Medien*, 5.–9.11.1997 Dessau, Stiftung Bauhaus Dessau, 1997, pp. 266–68.
- Snickars, Pelle, and Patrick Vonderau, editors. *The YouTube Reader*. National Library of Sweden, 2009.
- Soloway, Jill. »The Female Gaze.« *Topple Productions*, 11 Sept. 2016, <https://www.toppleproductions.com/tiff-master-class-the-female-gaze>.
- Stamps, Kelly. *How I Save Money at 23*. 2020, <https://www.youtube.com/watch?v=o2nHFEtuOlg>.
- . *My Packages Keep Getting Stolen Bc I Live in the Cut*. 2020. *YouTube*, <https://www.youtube.com/watch?v=tcT2FnnV8q8>.
- Stavans, Ilan. *I Love My Selfie*. Duke University Press, 2017.
- Thylstrup, Nanna Bonde, editor. *Uncertain Archives – Critical Keywords for Big Data*. The MIT Press, 2021.
- Titlow, John Paul. »To Take On HBO And Netflix, YouTube Had To Rewire Itself.« *Fast Company*, 15 May 2015, <https://www.fastcompany.com/3044995/to-take-on-hbo-and-netflix-youtube-had-to-rewire-itself>.
- Tyżlik-Carver, Magda. »| Curator | Curating | the Curatorial | Not-Just-Art Curating – A Genealogy of Posthuman Curating.« *Springerin | Hefte Für Gegenwartskunst*, Jan. 2017, <https://www.springerin.at/en/2017/1/kuratorin-kuratieren-das-kuratorische-nicht-nur-kunst-kuratieren/>.
- Universal Everything. »Future You.« *Universal Everything*, 2019, <https://universaleverything.com/projects/future-you>.
- Wang, Tricia. »Big Data Needs Thick Data.« *Ethnography Matters*, 13 May 2013, <http://ethnographymatters.net/blog/2013/05/13/big-data-needs-thick-data/>.
- Werner, Erich Alan. *Rants, Reactions, and Other Rhetorics – Genres of the YouTube Vlog*. University of North Carolina at Chapel Hill, 2012, <https://doi.org/10.17615/nna9-qz22>.
- Zappavigna, Michele. »The Organised Self and Lifestyle Minimalism – Multimodal Deixis and Point of View in Decluttering Vlogs on YouTube.« *Multimodal Communication*, vol. 8, no. 1, May 2019. DOI.org (Crossref), <https://doi.org/10.1515/mc-2019-0001>.
- Zhang, Hantian. »Evoking Presence in Vlogging – A Case Study of UK Beauty Blogger Zoe Sugg.« *First Monday*, Dec. 2017, <https://doi.org/10.5210/fm.v23i1.8107>.

## List of figures

Fig. 1: The overlapping of artistic and academic empirical approaches (author).	4
Fig. 2: <i>F0rwardpunk</i> . net.art project, 1999 (screenshot, author).	8
Fig. 3: <i>International Sputnik Day</i> . Installation and performance (digital prints, Fanzines, distributed online celebrations), installation view, Laden für Nichts, Leipzig, 2007 (Matthias Hennig).	8
Fig. 4: <i>The Setun Conspiracy</i> . Installation and performance (stanchions and walking line, cardboard sign, slides, cardboard office), installation view, Monitoring/Kunstverein Kassel, 2005 (Sven Heine).	9
Fig. 5: <i>History Exhaustion</i> . Installation (radio, antenna, sound, framed digital prints, book with narration, not depicted: three video performances), installation view, Galerie Metro, Berlin, 2009 (author).	9
Fig. 6: Elle Mills on Youtube.com (Screenshot, June 2018).	17
Fig. 7: Selection of YouTube videos by Anna's Analysis (screenshot, author).	26
Fig. 8: Selection of YouTube videos by Kelly Stamps (screenshot, author).	26
Fig. 9: Selection of YouTube videos by Markiplier (screenshot, author).	27
Fig. 10: Selection of YouTube videos by Marques Brownlee (screenshot, author).	27
Fig. 11: Selection of YouTube videos by Tara Brabazon (screenshot, author).	28
Fig. 12: Selection of YouTube videos by Casey Neistat (screenshot, author).	34
Fig. 13: Selection of YouTube videos by Elle Mills (screenshot, author).	34
Fig. 14: Selection of YouTube videos by Sailing La Vagabonde (screenshot, author).	35
Fig. 15: Neistat's video <i>Nintendo Switch on an airplane</i> (Neistat 2017) is entirely constructed of swish pan transitions.	39
Fig. 16: Drone flight sequence. The camera moves seamlessly from the ocean towards the harbor and the city (Carasu and Whitelum 2017).	40
Fig. 17: Hygge style – calm, relaxed, and pastel (Reid 2019b)	42
Fig. 18: Attention-grabbing style – rich, colorful videography (Mills 2018a).	42
Fig. 19: The <i>Data Proxy</i> outfit, with an orange-brimmed hat and expensive coffee (Photo: Walther Le Kon, with permission).	46
Fig. 20: Another variation of the outfit. (Photo: Walther Le Kon, with permission).	46
Fig. 21: Video editing software Final Cut Pro – User interface with timeline of Episode 6 (screenshot).	49
Fig. 22: Partial Screenshot of the <i>Data Proxy</i> website's homepage at <a href="http://www.dataproxy.biz">http://www.dataproxy.biz</a> with the lower end cropped (author, screenshot)	50

# Dank

Viele Menschen haben zu dieser Dissertation beigetragen und ich möchte meinen tiefsten Dank zum Ausdruck bringen.

Ich danke meinen Mentor\*innen Gabriele Schabacher (Johannes-Gutenberg-Universität Mainz), Henning Schmidgen (Bauhaus Universität Weimar), Yvonne Wilhelm (Zürcher Hochschule der Künste) und Christian Hübler (Zürcher Hochschule der Künste) für die fortwährende Unterstützung. Ihre Fragen, Anmerkungen, Lektüren, Vorschläge, E-Mails, Gutachten und grundlegendes und ehrliches Interesse an meiner Dissertation waren ein großer Ansporn.

Den Lehrenden, Kolleg\*innen und Peers möchte ich für ihr Feedback und die vertiefenden Gespräche danken. Es war mir eine große Freude und ein fortwährender Anlass miteinander ins Gespräch zu kommen: Magda Tyzlik-Carver, Andreas Broeckmann, Jürgen Winkler, Thomas Haigh, Marcus Burkhardt, Sebastian Gießmann, Christoph Engemann, Andreas Siekmann. Ich danke den Teilnehmer\*innen des Graduiertenkolloquiums Medienkulturwissenschaften Universität Mainz, die, neben den Mentor\*innen, zu den intensivsten Diskutanten der Textentwürfe zählen.

Ich danke den Lehrenden und den Ph.D.-Studierenden der Studiengänge Kunst und Design an der Bauhaus-Universität Weimar, vor allem Alex Toland, Alexander Schwinghammer, Frank Hartmann, Michael Lüthy, sowie Maria Linares, Edith Kollath, Markus Schlaffke, Katja-Marie Voigt, Johanna Hodde, Emanuel Mathias, Anne Brannys, Gabriel Ben Moshe, Evelina Rajica und des weiteren Ferdinand Ulrich, Michael Heidt und Malte Westphalen.

Der künstlerische Teil wurde durch Ausstellungen und Workshops angereichert und ich möchte dafür namentlich Lena Brüggemann, Sabine Winkler, Vera Tollmann, David Riff, Ekaterina Degot, Mark Tribe, Anneliese Ostertag, Frank Motz, Michael Arzt, Olia Lialina, Janos Sugar und Sakrowski für deren Ermöglichung danken. Ohne Marie-Eve Levasseur, !Mediengruppe Bitnik, Joanna Bryson, Walther Le Kon, Greta & Marvin, Tega Brain & Surya Mattu wäre der Data Proxy farblos geblieben.

Ich danke meiner Partnerin Claudia Lindner, die den Balance-Akt zwischen konzentriertem Rückzug und Bewältigung der Alltagsdinge mit mir geduldig und kreativ absolvierte. Meinen Töchtern Ada und Marie gelang es, mich einerseits zu erden und andererseits in Staunen darüber zu versetzen, wie sehr das Lernen Menschen menschlich macht. Dem Ex-Robotroner Heinz Lindner verdanke ich zahlreiche inhaltliche Anmerkungen. Uschi Hunger unterstützte mich durch die konzentrierte und unermüdliche Durchsicht des Manuskripts. Anne, Du bist ein Schatz! Hans, der den Anfang der Dissertation mitverfolgt hat, verabschiedete sich deutlich zu früh von dieser Erde. Cornelia Friederike Müller, Rene Pölzing und Matthias Hennig inspirieren mich fortwährend.

Für die ideelle Unterstützung und die jahrelange großartige Zusammenarbeit danke ich zutiefst Inke Arns beim Hartware MedienKunstVerein und Anna Berkenbusch, sowie dem ehemaligen Kollegium des Studienganges Kommunikationsdesign/Burg Giebichenstein. Ein Teil meiner Forschung wurde durch ein Graduiertenstipendium des Landes Thüringen unterstützt, welches ein konzentriertes Arbeiten ermöglichte.

Ich danke den Interviewpartner\*innen für Ihre großzügige Auskunft zu zahlreichen Details: Jürgen Bittner, David Childs, Sharon Codd, Birgit Demuth, Rolf Heinemann, Leander Seige, Matthias Uhlig, Uwe Wloka, Michael Zipf und mehrere, ehemalige SAP-Mitarbeiter\*innen, sowie dem Archivar

Wolfgang Wimmer. Ich danke den dem Rechenwerk Computer- & Technikmuseum Halle, insbesondere Constanze Czech.

Abschließend sei all jenen gedankt, welche die Infrastrukturen und Bürokratien aufrechterhalten, die dieses Unterfangen überhaupt ermöglichen: Den Administrator\*innen, den Bibliothekar\*innen in Weimar und Leipzig, den Universitätsmitarbeiter\*innen, dem Team der Bauhaus Research School, den Fördermittelgeber\*innen, den Bahnmitarbeiter\*innen, den illegalen PDF-Anbieter\*innen, der Belegschaft meines Stamm-Cafés und den Datenbankadministrator\*innen dieser Welt.