PROCEEDINGS

53. IWK

Internationales Wissenschaftliches Kolloquium
International Scientific Colloquium

Faculty of
Mechanical Engineering

PROSPECTS IN MECHANICAL ENGINEERING

8 - 12 September 2008

www.tu-ilmenau.de

TECHNISCHE UNIVERSITÄT
ILMENAU

Home / Index:
http://www.db-thueringen.de/servlets/DocumentServlet?id=17534

Published by
Impressum

Publisher Herausgeber	Der Rektor der Technischen Universität Ilmenau Univ.-Prof. Dr. rer. nat. habil. Dr. h. c. Prof. h. c. Peter Scharff
Editor Redaktion	Referat Marketing und Studentische Angelegenheiten Andrea Schneider Fakultät für Maschinenbau Univ.-Prof. Dr.-Ing. habil. Peter Kurz, Univ.-Prof. Dr.-Ing. habil. Rainer Grünwald, Univ.-Prof. Dr.-Ing. habil. Prof. h. c. Dr. h. c. mult. Gerd Jäger, Dr.-Ing Beate Schlütter, Dipl.-Ing. Silke Stauche
Editorial Deadline Redaktionsschluss	17. August 2008
Publishing House Verlag	Verlag ISLE, Betriebsstätte des ISLE e.V. Werner-von-Siemens-Str. 16, 98693 Ilmenau

CD-ROM-Version:

Implementation Realisierung	Technische Universität Ilmenau Christian Weigel, Helge Drumm
Production Herstellung	CDA Datenträger Albrechts GmbH, 98529 Suhl/Albrechts

ISBN: 978-3-938843-40-6 (CD-ROM-Version)

Online-Version:

Implementation Realisierung	Universitätsbibliothek Ilmenau ilmedia Postfach 10 05 65 98684 Ilmenau

Home / Index:
http://www.db-thueringen.de/servlets/DocumentServlet?id=17534

53rd Internationales Wissenschaftliches Kolloquium
Technische Universität Ilmenau
08 – 12 September 2008

St. Waßmuth / M. Dambon / G. Linß

Customised Software Tools for Quality Measurement – Application of Open Source Software in Education

Introduction

The Free and Open Source Software Movement is currently the most interesting and influential trend in software industry as it enables nearly boundless access to software [1, 2]. On the other side future generations of engineers will require more profound software knowledge then today. As it could be ascertained, most students still have deficiency in software knowledge. That implies the capability to describe a problem in an abstract manner in order to be later implemented easier into software as well as to be well versed with a sort of software tools which are considered as elementary. The necessary knowledge of a certain programming language can thereby differ from case to case.

University education has it strength in teaching theory and methodical understanding. Nevertheless practical relevance is sometimes absent or acquired knowledge could not be adopted and therefore is forgotten promptly. Through application of software in higher education students can both adopt and enhance acquired knowledge adequately. Beside students can amplify their software skills for later job.

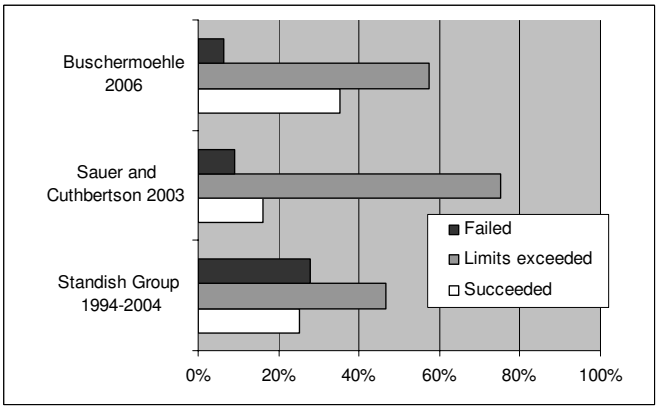

Fig. 1 – Outcomes of surveys investigating software projects in enterprises [3, 4, 5]

The lack of software knowledge in industry today is shown in a variety of international surveys [3, 4, 5]. Accordingly, most software projects in enterprises are failing. Either they are cancelled or they are completed at the cost of budget or time overrun or with a reduced functional scope (fig.1).

Furthermore studies on the application of software for quality assurance in Germany show that only one third of all users purchase commercial off-the-shelf software (COTS) for quality assurance [6, 7]. Most companies either disclaim the application of any specialised software solution or develop it by their own. The latter are often based on the Microsoft Office package [6].

The British Computer Society (BSC), referring to the mentioned deficits in industry, states: “Improving education will not make an immediate change to practice, but is a vital part of a long-term solution to the problem…” [8].

Therefore a new level in education has been developed and applied at the Technische Universität Ilmenau. Based on a SQL database server and Office clients from Microsoft Office and OpenOffice.org students in mechanical engineering are enabled to develop customised software solutions to problems of practical relevance without having any specific knowledge in advance. Applied Open Source Software tools made development process easier and software solutions more powerful on the other hand. The following objectives should be accomplished:

- Enhance knowledge by practical experience,
- Gaining understanding about the process of software development and architecture of multi-layer software systems,
- Learn the handling of important standard software for later job,
- Training of presentation and teamwork.

From the scientific point of view, the frequent problems at the development of software and the evaluation of quality of software solutions were of particular importance.

Approach

The extent of selected Open Source Software for education purpose is demonstrated in fig. 2 and shows a viable software architecture which also could be deployed in industry as a cost efficient alternative to Commercial Software solutions.

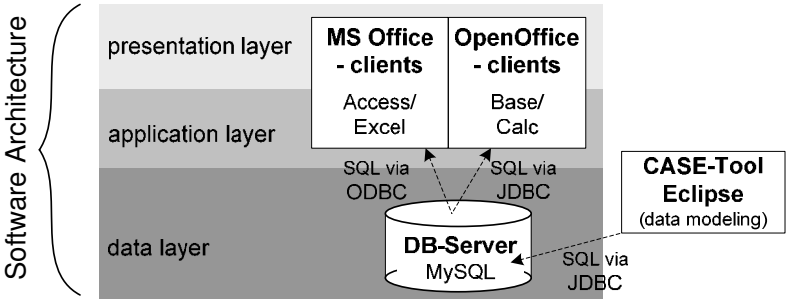

Fig. 2 – Selected software components

All chosen components except Microsoft Office client – are popular Open Source Software: an Eclipse plug-in [9] as CASE-tool (Computer Aided Software Engineering) for data modelling, MySQL [10] as database system and the OpenOffice.org client [11]. Connection between the software components had been made through SQL via JDBC and ODBC middleware.

Based on the fact of a virtual enterprise, students had the task to develop small software modules to be applied in metrology and quality assurance. To build up a complex software system it is necessary to apply a systematic approach. Figure 3 shows the adopted approach containing all typical main steps of software lifecycle as well as selected technology [5].

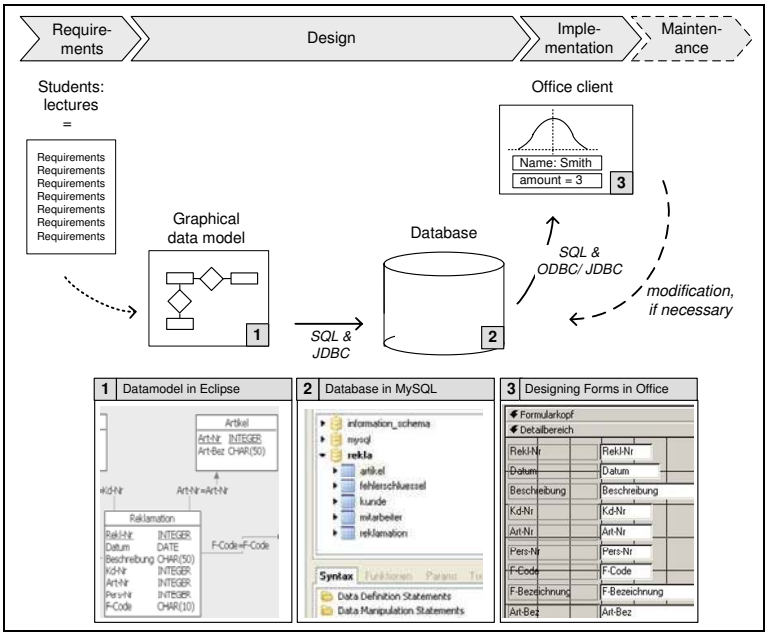

Fig. 3 – Process and tools applied for students software tools

Every group was made up of two or three students and had to select a typical field of interest in quality assurance to be later realised with software. At the beginning each student had to give a lecture which should include the theoretical basics as well as a plausible example to the other students. The lecture was also seen as the requirements document and the result of requirements analysis.

Results

With one exception all groups had shown viable Office clients using Microsoft Excel/ OpenOffice Calc and/or Microsoft Access/ OpenOffice Base. (Note: Initially only Microsoft Office package had been used. From the second semester on also

OpenOffice.org had been applied.)

Typical software tools which had been realized are, among others:

- Machine and Process Capability Analysis,
- Statistical Process Control (Control Charts),
- Acceptable Quality Level (AQL) Sampling Inspection,
- Supplier Assessment,
- Complaints Management,
- Failure Mode and Effects Analysis (FMEA),
- …

Example: Statistical Process Control (SPC)

The purpose of this self-developed SPC-tool is to monitor a process routinely through the use of attribute control charts to measure failure per sample (binomial distribution) and failure per probe unit (Poisson's distribution). The program was solely realised with MS Excel connected to the MySQL database server containing the data of characteristics and the measuring data. At the beginning user has to choose the type of control chart to use and the characteristic to be measured. Afterwards the program retrieves the measuring data from database and calculates corresponding upper/ lower control and action limit and finally plots the chart. Additionally user can add more data and save them in database or switch display between 'order by time' or 'order by number', fig. 4.

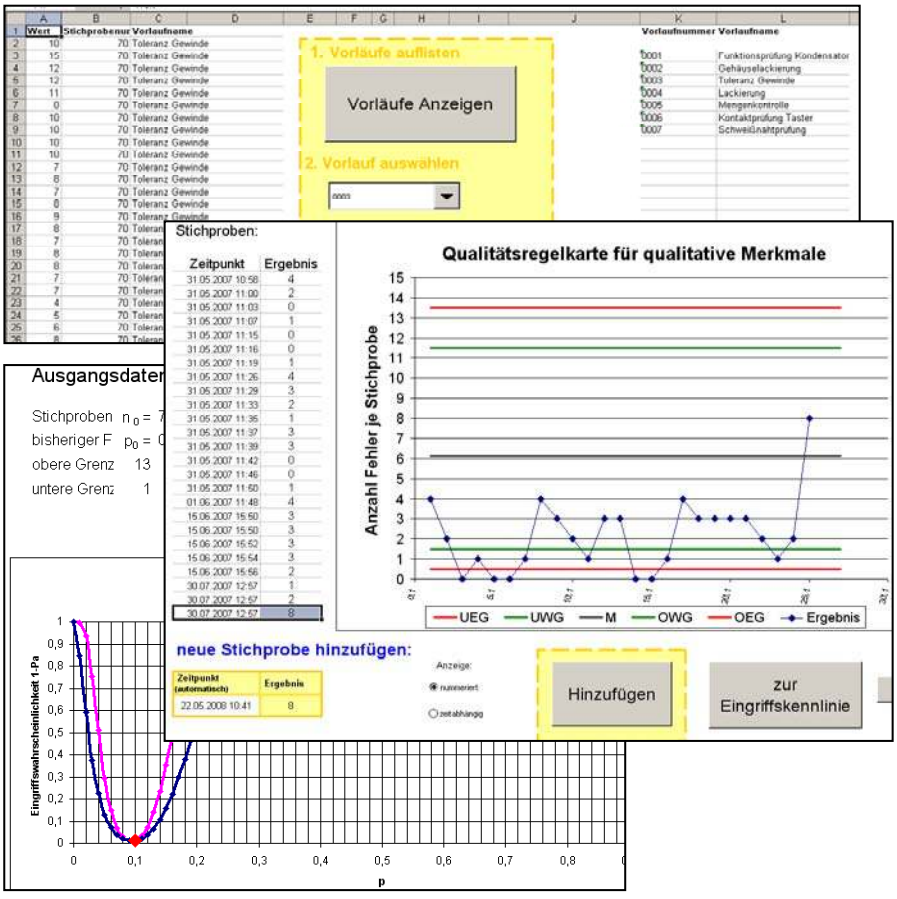

Fig. 4 – Example of SPC-Tool made by students with MS Excel.

From 68 students who attended the courses at the beginning only 47 passed it. Those who changed (all within the first three weeks) reclaimed the additional effort regarding to a 'normal' course. At the beginning and at the end of the semester a standardised questionnaire was handed out to each student. One intention was to evaluate student's knowledge concerning Microsoft Office/ Open Office and databases, both through self assessment and control questions. Figure 5 shows a good correlation and an overall remarkable increase in knowledge.

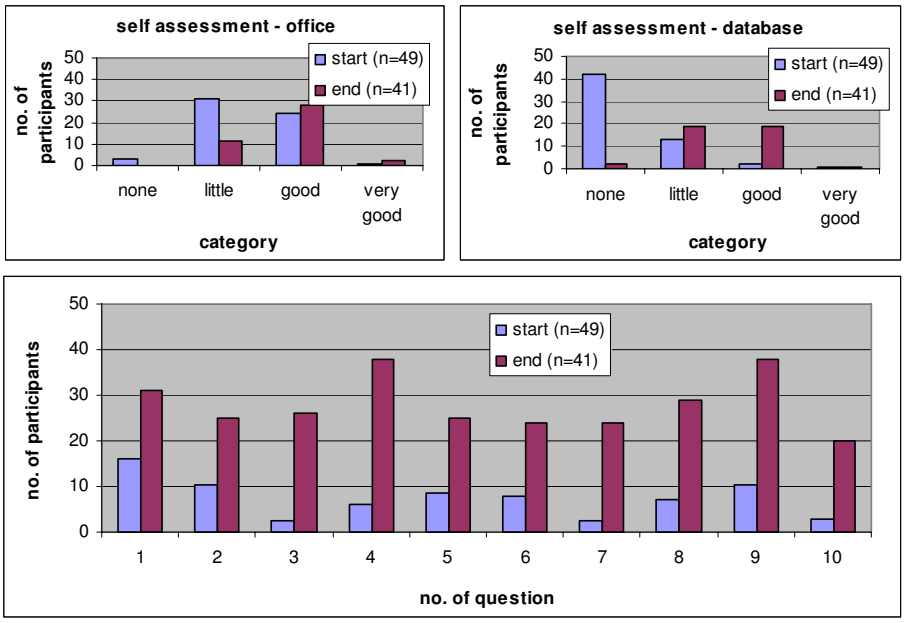

Fig. 5: Increased knowledge, above self assessment – below control questions (n=41, note: six forms were not applicable)

As it has been outlined, students were able:

- To create small individualized software applications for quality assurance using Office clients and a SQL database system,
- Without any previous software knowledge (fig. 5),
- And to develop and customise it by there own in a relatively short time – a necessary attribute for effective working with software.

During development every group became well aware of all those problems and necessary steps which have to be overcome and are so typical in every major software project. This training approach had been enabled only through broad application of Open Source software tools, rapid testing and customizing as well as through good software documentation on the Internet and program assistance within the tools.

Most developed software solutions were limited to the initially proposed course examples instead of covering a more general application range. This was mainly due to limitations of the underlying data model. Developing high quality data models needs both comprehensive expertise and data modelling understanding [12]. Nevertheless the majority of all participants were quite enthusiastic regarding the new approach in education compared to standard courses, fig. 6. Interestingly nearly one half of the students were quite sceptical at the beginning of the course

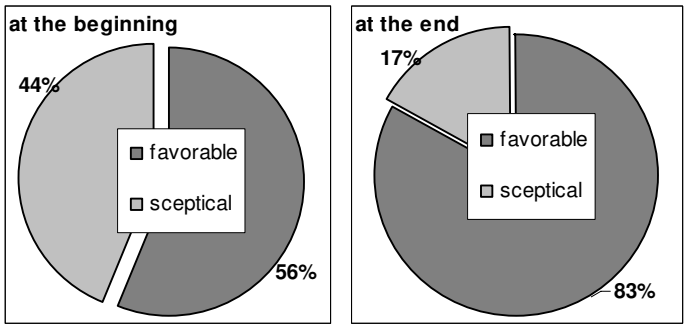

Fig. 6 – Attitude to course at the beginning and at the end (n=41)

References:

[1] Balduzzi, M.: On the Influence of Free Software on Code Reuse in Software Development http://www.idi.ntnu.no/emner/dt8100/Essay2005/balduzzi.pdf [08-06-10]

[2] David A. Wheeler: Why Open Source Software / Free Software (OSS/FS)? Look at the Numbers! www.dwheeler.com/oss_fs_why.html [08-06-10]

[3] Buschermöhle, R.; u. a.: Success – Erfolgs- und Misserfolgsfaktoren bei der Durchführung von Hard- und Software-entwicklungsprojekten, 2006, http://vsek01.informatik.uni-oldenburg.de/ ~joomla [08-06-10]

[4] The Standish Group International, 1994-2004, http://www.standishgroup.com/index.php [08-06-10]

[5] Sauer, C.; Cuthbertson, C.: The State of IT Project Management in the UK, 2003, http://www.bestpracticehelp.com/The%20State%20of%20IT%20Project%20Management%20in%20the%20UK%202003-2004.pdf [08-06-10]

[6] Waßmuth, St.; Dambon, M.; Linß, G.: Anwendungsentwicklung für den CAQ-Bereich – Ergebnisse eines neuartigen Ausbildungskonzeptes an der TU Ilmenau. In: Innovationsqualität: Qualitätsmanagement für Innovationen: Bericht zur GQW-Tagung 2008 - Bremen / Gert Goch (Hrsg.) - Aachen: Shaker, 2008, p.79-101

[7] Fuchs, J.; Krautwasser, St.: Falsch geschätzt – Trendstudie: Wie viel Software braucht das Qualitätsmanagement? In: Qualität und Zuverlässigkeit QZ, München: Hanser, 50, p.20-23, 2005

[8] Report of a working group from The Royal Academy of Engineering and The British Computer Society: The Challenges of Complex IT Projects, 2004, http://www.bcs.org/upload/pdf/complexity.pdf [08-06-10]

[9] Homepage of the Eclipse Project and the Azzurri Data Modelling Plug-in www.eclipse.org and www.azzurri.jp/en/software/clay/ [08-06-10]

[10] Homepage of MySQL www.mysql.com [08-06-10]

[11] Homepage of OpenOffice.org http://www.openoffice.org [08-06-10]

[12] Ponniah, P.: Data modelling fundamentals - a practical guide for IT professionals, Hoboken: Wiley: 2007

Authors:

Univ.-Prof. Dr.-Ing. habil. Gerhard Linß, Dipl.-Ing. Martin Damon, Dipl.-Wirtsch.-Ing. Stefan Waßmuth
TU Ilmenau, Department of Quality Assurance, Faculty of Mechanical Engineering,
Gustav-Kirchhoff-Platz 2
98693, Ilmenau
Phone: +49 3677/ 69-3937
Fax: +49 3677/ 69-3823
E-mail: stefan.wassmuth@tu-ilmenau.de