

Mäder, Patrick:

**The benefits of traceability during software implementation**

**DOI:** [10.22032/dbt.42351](https://doi.org/10.22032/dbt.42351)

**URN:** [urn:nbn:de:gbv:ilm1-2020200337](https://nbn-resolving.org/urn:nbn:de:gbv:ilm1-2020200337)

---

*Original published in:* Software and systems traceability for safety-critical projects : report from Dagstuhl Seminar 15162 / Cleland-Huang, Jane; Rayadurgam, Sanjai; Mäder, Patrick; Schäfer, Wilhelm. - Wadern : Schloss Dagstuhl. - 5 (2015), 4, p. 85-86.  
(Dagstuhl Reports : Dokumentationen zu Dagstuhl-Seminaren und Dagstuhl-Perspektiven-Workshops / Schloss Dagstuhl, Leibniz-Zentrum für Informatik. - Wadern : Schloss Dagstuhl. - 5 (2015), 4, p. 76-97. - DOI: [10.4230/DagRep.5.4.76](https://doi.org/10.4230/DagRep.5.4.76))

*ISSN:* 2192-5283

*Original published:* 2015-12-21

*[Visited:* 2020-03-12]



This work is licensed under a [Creative Commons Attribution 3.0 Unported](https://creativecommons.org/licenses/by/3.0/) license. To view a copy of this license, visit <https://creativecommons.org/licenses/by/3.0/>

---

### 3.8 The Benefits of Traceability During Software Implementation

*Patrick Maeder (TU Ilmenau, DE)*

**License** © Creative Commons BY 3.0 Unported license  
© Patrick Maeder

**Joint work of** Maeder, Patrick; Egyed, Alexander

**Main reference** P. Mäder, A. Egyed, “Do developers benefit from requirements traceability when evolving and maintaining a software system?”, *Empirical Software Engineering*, 20(2):413–441, 2015.

**URL** <http://dx.doi.org/10.1007/s10664-014-9314-z>

Software traceability is a required component of many software development processes. Advocates of software traceability cite advantages like easier program comprehension and

support for software maintenance (i.e., software change). However, despite its growing popularity, for a long time there existed no published evaluation about the usefulness of requirements traceability. It is important, if not crucial, to investigate whether the use of requirements traceability can significantly support development tasks to eventually justify its costs [3, 1, 4, 5]. We thus conducted a controlled experiment with 71 subjects re-performing real implementation tasks on two third-party development projects: half of the tasks with and the other half without traceability. Our findings show that subjects with traceability performed on average 24% faster on a given task and created on average 50% more correct solutions [2, 6] – suggesting that traceability not only saves effort but can profoundly improve software implementation quality. For a follow-up study [7], we selected medium to large-scale open-source projects and focused especially on the discovered effect for implementation quality. We quantified for each developed component of each software project, the degree to which a studied development activity was enabled by existing traceability and set this metric in relation to the number of defects that occurred in a component. We found that traceability significantly affects the defect rate in a component. Overall, our results provide for the first time empirical evidence that traceability significantly improves implementation speed as well as implementation quality during software development.

#### References

- 1 P. Mäder, O. Gotel, and I. Philippow (2009). “Motivation matters in the traceability trenches”, In: *Proc. of RE’09*, pp. 143–148.
- 2 P. Mäder and A. Egyed (2012). “Assessing the effect of requirements traceability for software maintenance”, In: *Proc. of ICSM’12*, pp. 171–180.
- 3 E. Bouillon, P. Mäder, I. Philippow (2013). “A Survey on Usage Scenarios for Requirements Traceability in Practice”, In: *Proc. of REFSQ’13*, pp. 158–173.
- 4 P. Rempel, P. Mäder, and T. Kuschke (2013). “An empirical study on project-specific traceability strategies”, In: *Proc. of RE’13*, 195–204.
- 5 P. Rempel, P. Mäder, T. Kuschke, and I. Philippow (2013). “Requirements Traceability across Organizational Boundaries – A Survey and Taxonomy”, In: *Proc. of REFSQ’13*, pp. 125–140.
- 6 P. Mäder and A. Egyed (2015). “Do developers benefit from requirements traceability when evolving and maintaining a software system?”, In: *Empirical Software Engineering*, 20(2), pp. 413–441.
- 7 P. Rempel and P. Mäder (2015). “Estimating the Implementation Risk of Requirements in Agile Software Development Projects with Traceability Metrics”, In: *Proc. of REFSQ’15*, pp. 81–97.