



Call for Papers: Seventh Workshop on Managing Technical Debt (MTD 2015)

To be held with ICSME 2015 October 2 in Bremen, Germany

Delivering complex, large-scale systems faces the ongoing challenge of how best to balance rapid deployment with long-term value. Theoretical foundations and empirical evidence for analyzing and optimizing short-term versus long-term goals in large-scale projects are needed. From the original description—"not quite right code, which we postpone making right"—various people have used the metaphor of technical debt to describe many kinds of debts or ills of software development. On one hand, the practitioner community has increased interest in understanding and managing debt. On the other hand, the research community has an opportunity to study this phenomenon and improve the way it is handled. We can offer software engineers a foundation for managing such tradeoffs based on models of their economic impacts. Technical debt succinctly communicates the issues observed in large-scale long-term projects:

- There is an optimization problem where focusing on the short-term puts the long-term into economic and technical jeopardy.
- Design shortcuts can give the perception of success until their consequences start slowing projects down.
- Software development decisions, especially architectural ones, must be actively managed and continuously analyzed quantitatively as they incur cost, value, and debt.

Yet many questions remain open, such as

- What is the lifecycle of technical debt?
- How is technical debt related to evolution and maintenance activities?
- How can information about technical debt be empirically collected for developing conceptual models?
- What metrics need to be collected so that key measurement and pay-off analysis can be conducted?
- How can technical debt be visualized and analyzed?
- How should we manage technical debt incurred by external business constraints such as acquisitions and market ecosystems?
- How can we assign business value to intrinsic qualities (e.g., cohesion and coupling)?
- How do we manage dependencies between different items of technical debt?
- How can we create traces between technical debt items and other software engineering artifacts?
- How can we quantify costs and benefits of refactorings?
- What are the right tools for managing technical debt?
- How can we apply financial theories to manage technical debt?
- How can we benchmark the tools that identify and measure technical debt?

We seek papers on practical experience with technical debt and approaches that attempt to answer these questions. All submitted papers must conform to the IEEE Conference Publishing Services (CPS) [formatting instructions](#), and must not exceed 8 pages for all text, inclusive of figures, tables, and appendices, with up to one additional page for references only. All submissions must be in PDF. Submit your paper electronically via [EasyChair](#). We invite submissions of papers in any areas related to the themes and goals of the workshop in the following categories:

1. Research papers, describing innovative and significant original research in the field
2. Industrial papers, describing industrial experience, case studies, challenges, problems, and solutions

In either category, we look for long papers (8 pages), describing mature results, and short papers (4 pages), describing emerging results and future trends.

Important dates:

- Paper submissions: June 12, 2015
- Notification of authors: July 3, 2015
- Camera-ready copies: July 24, 2015
- Workshop: October 2nd, 2015

Organizers:

Neil Ernst, Carnegie Mellon Software Engineering Institute
Paris Avgeriou, University of Groningen
Philippe Kruchten, University of British Columbia

More information: <http://www.sei.cmu.edu/community/td2015/>