TechDebt 2018: The Journey from Metaphor to Theory and Practice

Robert L. Nord

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Document Markings

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM18-0687

What is Technical Debt?

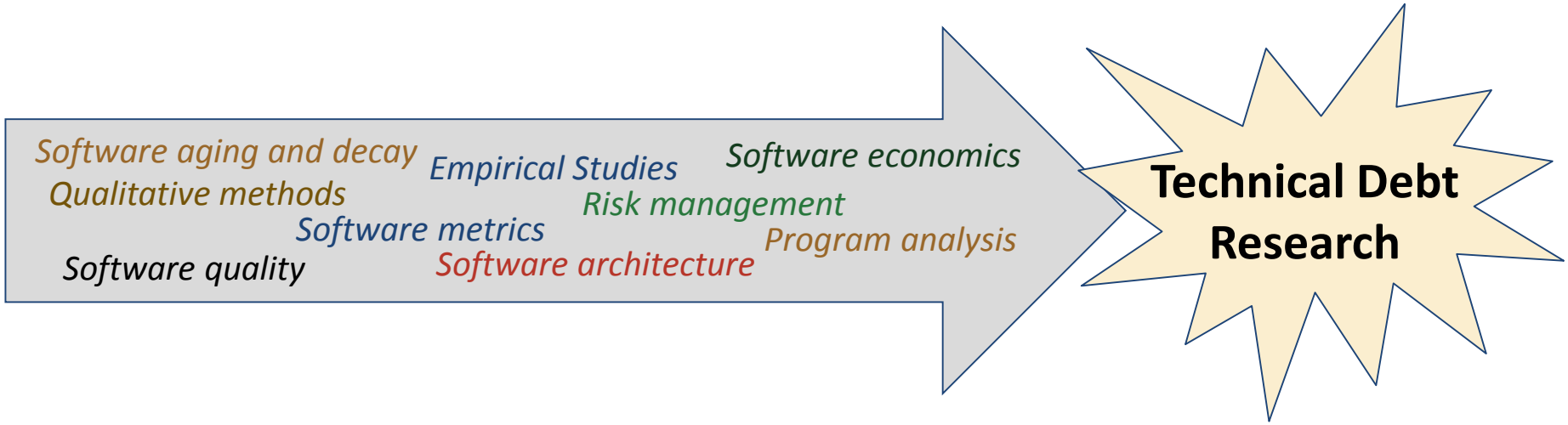
In software-intensive systems, technical debt is the collection of design or implementation constructs that are expedient in the short term, but sets up a technical context that can make future changes more costly or impossible.

Technical debt presents an actual or contingent liability whose impact is limited to internal system qualities, primarily maintainability and evolvability.

April 2016, Dagstuhl
http://mtd2016dagstuhl.org

Evolution and Convergence

Some 30 years of R&D in software engineering:

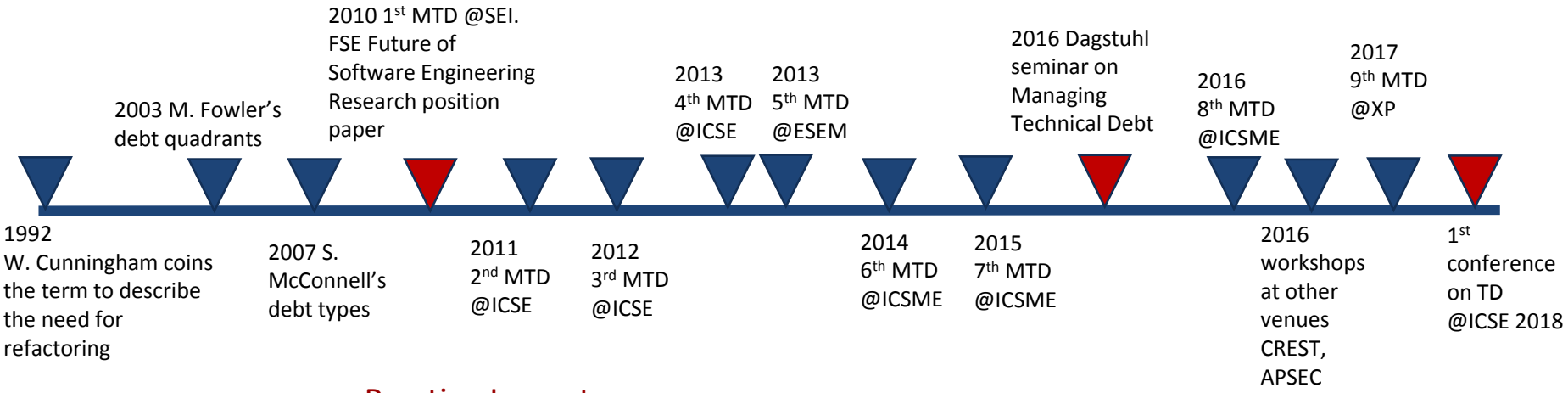


Seaman 2013

Our Journey

Research Impact

Over 200 research papers and many blog posts
MTD 2011-2016: 70 papers, 1394 citations
Special issues: IEEE Software and JSS
Systematic mapping studies

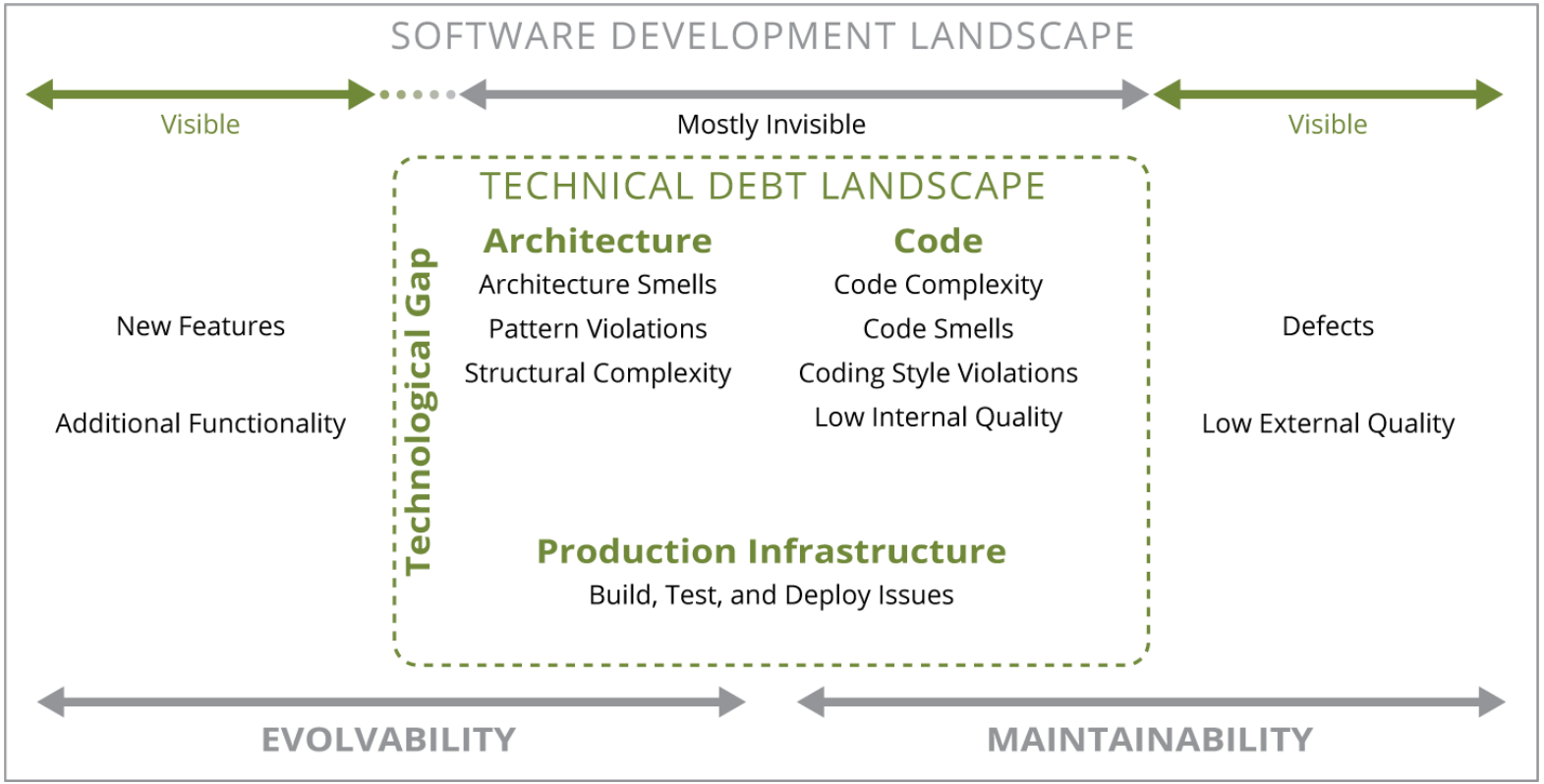


Practice Impact

Organizations looking into developing technical debt practices
Tool vendors repurposing tools to detect technical debt
OMG Specification: Automated technical debt measures

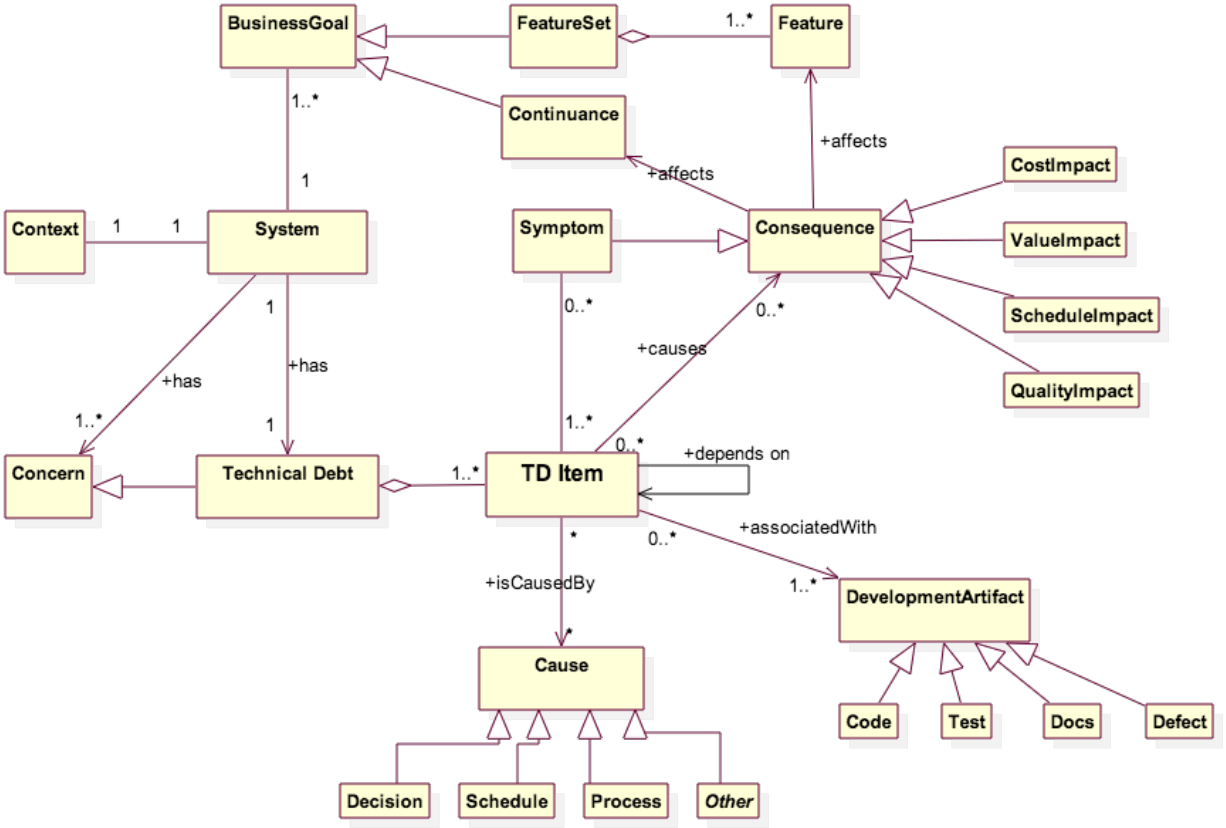
Previous Editions at http://www.techdebtconf.org

Technical Debt Landscape



Adapted from Kruchten, P. Nord, R.L., Ozkaya, I. 2012. Technical Debt: From Metaphor to Theory and Practice, IEEE Software, 29(6), Nov/Dec 2012.

Conceptual Model of Technical Debt

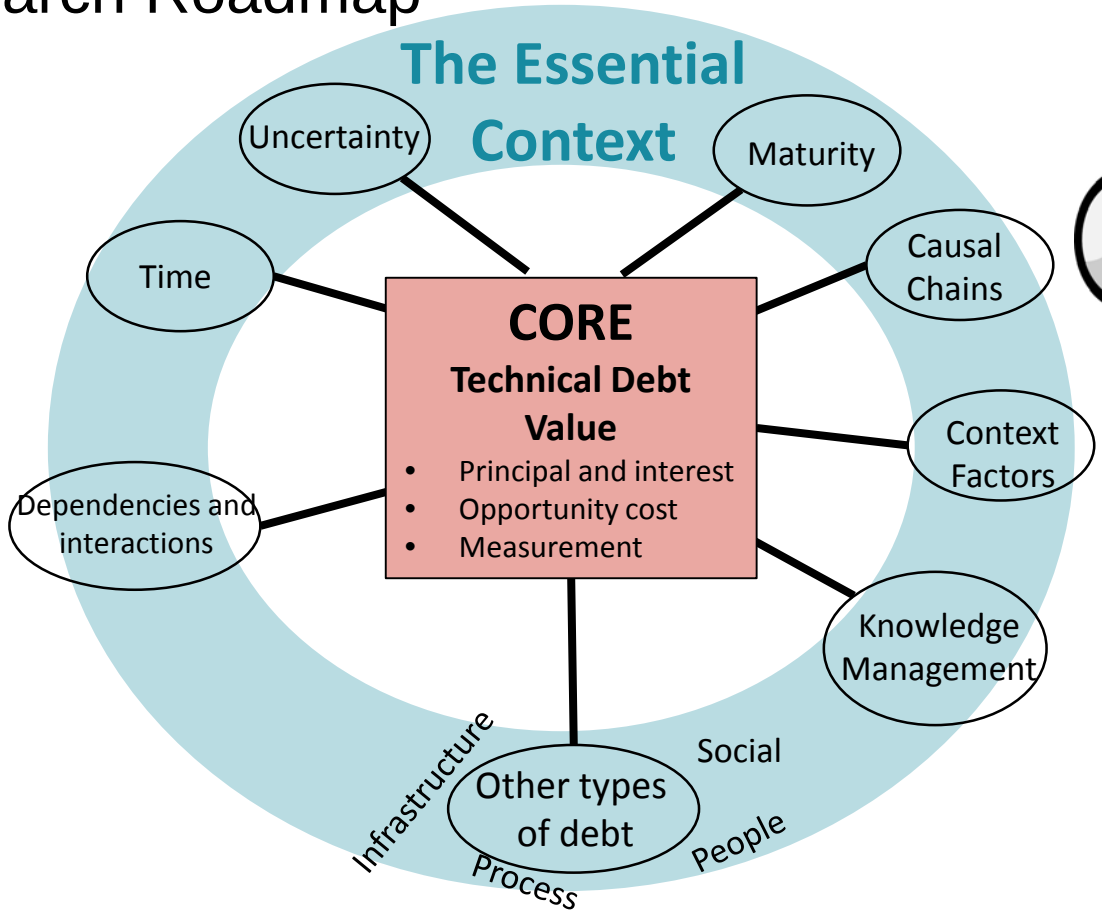


Vision for Managing Technical Debt

Challenge	Initial Step	Vision
Process and tools	Bring visibility to technical debt. Strategies include tagging issues, allocating resources, and reflecting during retrospectives.	Processes and tools manage technical debt holistically throughout the lifecycle.
Software economics	Track consequences of tradeoffs in terms of business and economic impact.	Technical debt management is used as a strategic software development approach.
Software architecture	Use software architecture during development, testing, and operations by managing quality attribute issues related to technical debt.	Technical debt is managed at the level of architecture decisions and associated tradeoffs and risks.
Empirical basis and data science	Enable data collection in development activities. Such information includes iteration tempo, defect rate, files changed frequently.	Software development data is used for technical-debt analysis and provide an empirical basis for decision making.
Education	Instruct how to avoid technical debt and how to use quality management tools and architecture reviews to uncover technical debt.	Technical debt is an integral part of the curriculum, not a separate course but a learning thread permeating course work.

Avgeriou, P., Kruchten, P., Nord, R., Ozkaya, I., and Seaman, C. 2016. Reducing Friction in Software Development. IEEE Software 33, 1 (2016), 66−73.

Research Roadmap



Our Goals Today

Collectively focus on the following to contribute to a roadmap for progress

- Most pressing industry problems
- Most promising research approaches
- Hard research questions
- The necessary infrastructure

Agenda

Social media hash tag: #techdebt18

Keynote: Eoin Woods, Endava

- *The Past, Present, and Future of Technical Debt*

Four paper sessions

- *Incurring Debt*
- *Assessing Debt*
- *Managing the Debt II*
- *Practice in Industry*

Two working sessions with short presentations and discussion

- *Managing the Debt I*
- *Future Research*

Tools session

Logistics

Two breaks at 10:30 and 3:30

Lunch at 12:30-2:00

Dinner at 6:00-10:00

Join our mailing list (see link at):

http://www.techdebtconf.org

Dinner @ Gothia Towers Seasons, 6:00 pm – 10:00 pm

Mushroom

Mushrooms on toast, smoked brisket of beef, Västerbotten cheese crème, dried lingonberries, cress and local aged cheese

Cod

Potato and cauliflower croquette, blackened carrots, roasted cavolo nero and shellfish velouté flavoured with dill crown

Lingonberry parfait

Almond crisp, lemon and liquorice meringue, lingonberry crème, lingonberry jam and azina cress

Gothia's choice of white wine 1 glass/person
Gothia´s choice of red wine 1 glass/person with a refill
Dessert wine 6 cl/ person

Coffee/tea

Quick round among all participants

Please introduce yourself and state your personal expectations for the conference!

Backup Slides

Vision

TD will be managed as well as we now manage defects and new features

We have a clear, operational definition of “good enough”

We have a way to translate developer concerns to manager concerns

TD will be incurred intentionally most of the time

Projects that manage TD are efficient, effective and sustainable

The benefit of upfront architectural work (vs. emergent architecture) is proven

Tools support all aspects of TD management that are used by all stakeholders

TD-aware development is an accepted way of producing software

Architectural assessment is part of policy

Plans for the Next Conference

International Conference on Technical Debt (TechDebt 2019)

Held in conjunction with the International Conference on Software Engineering (ICSE).

Tentative date: May 26-27, 2019

Conference organizers

- Ipek Ozkaya, Software Engineering Institute

Contact Information

Robert L. Nord
Architecture Practices Initiative
Email: rn@sei.cmu.edu

U.S. Mail
Software Engineering Institute
Customer Relations
4500 Fifth Avenue
Pittsburgh, PA 15213-2612
USA

Web
www.sei.cmu.edu
www.sei.cmu.edu/architecture

Customer Relations
Email: info@sei.cmu.edu
SEI Phone: +1 412-268-5800
SEI Fax: +1 412-268-6257