
On the Limits of the Technical Debt Metaphor

Some Guidance on Going Beyond

Prof. Dr. Klaus Schmid
SSE, University of Hildesheim, Germany
schmid@sse.uni-hildesheim.de

On the notion of technical debt

- Technical debt as a metaphor
 - good analogy to encourage refactoring
 - .. but fundamentally flawed
- *Why technical debt isn't a debt*
 - It depends on the future – and what your future development items will be
- *If there is technical debt – what is a debt-free system?*
 - Only possible to answer in retrospect whether a system was “debt-free”
 - Even if we would know all possible futures, it would be nearly impossible to determine what would be the optimal system
 - Technical debt: strongly connected to evolution & maintainability

An important distinction

Potential technical debt:

a system structure that seems to be inadequate based on general criteria about good structuring of systems

Important:

potential technical debt = *risk*

effective technical debt = *problem*

Effective technical debt:

a system structure that leads to avoidable effort in future evolution

⇒ (Effective) Technical debt can only be determined :

- *In relation to future evolution*
- *Relative to an optimal implementation*
- *Which can be described formally:*

$$TD(S, e) = \max\{ CC(S, e) - CC(S', e) | S' \in Sys(S) \}$$

How to value technical debt:

- Many different ways
- *For some:* connections to TD are at best indirect
- Issues:
 - Additivity of valuations
 - Additivity of technical debt (*in general not*)
 - Options: magnify uncertainty

So far these issues seem to be addressed only in a rather limited way

Summary

- Lack of a precise definition of technical debt
 - Importance to include future
 - Differentiation between content and structure in evolution
- Distinction between
 - Potential
 - Effective
- Valuation issues
 - Problems in direct valuation
 - Problems in combining contributions

QUESTIONS?
COMMENTS?