Veracity Technology Spearhead

Enabling end-to-end veracity within value exchange ecosystems

Quantifying Veracity Requirements Debt in Software-based Technology

Judith Perera, Faith Culas, Natalia Ortega, Matthias Galster, Kelly Blincoe, Ewan Tempero, Yu-Cheng Tu

WHAT IS TECHNICAL DEBT?

Technical debt (TD) occurs when software teams make sub-optimal decisions to deliver a product soon.

Just like financial debt, technical debt accrues interest too.

Quantification of technical debt helps better management of it

Types of technical debt:

- Code
- Design
- Architecture
- Requirements
- Documentation
- Test

TD Management activities:

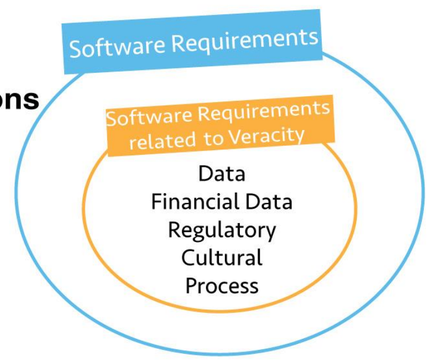
- Identification
- Measurement (Quantification)
- Prioritization
- Repayment
- Monitoring

You can get it done faster but you will pay later

Financial debt can be easily identified and quantified but technical debt is not easily quantifiable and can be easily neglected and lead to adverse consequences.

WHAT IS VERACITY DEBT?

The definition of veracity: "Trust, truth, authenticity, and demonstrability."
Veracity Requirements Technical Debt (VRTD) or Veracity Debt captures the consequences of such sub-optimal decisions made concerning veracity requirements.



Quantification provides a foundation to better manage VRTD in software projects.

FIVE TYPES OF VERACITY

1. **Data Veracity**
   attributes such as, provenance, transparency, sovereignty, integrity, accuracy, correctness of the data entered in software systems
2. **Financial Data Veracity**
   correctness of banking, tax, sales, and accounting data
3. **Regulatory Veracity**
   compliance with regulations, standards, and other regional, local, or international laws.
4. **Cultural Veracity**
   ensuring that information relating to provenance and cultural authorities is maintained in the digital records associated with the software systems
5. **Process Veracity**
   adherence to organizational policies, processes, methods, and practices

WHY QUANTIFY VERACITY DEBT?

To make informed management decisions and thus avoid possible adverse consequences

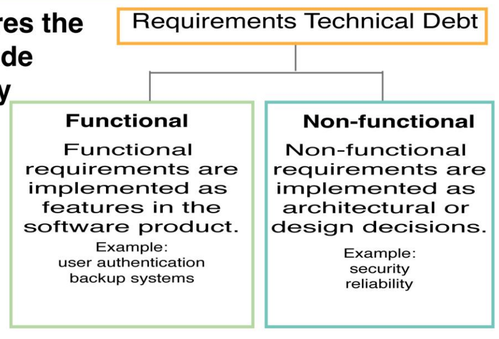
Possible consequences due to incorrect management of veracity debt

- **COST** for design/code rework
- **MISSING** essential user needs
- **COMPROMISE** of customer's data
- **REPUTATION LOSS** to company
- **LOSS** of customer's trust
- **REGULATORY FINES** to company
- **FINANCIAL LOSS** to company
- **DISCLOSURE** of trade secrets

WHAT IS REQUIREMENTS TECHNICAL DEBT?

"Requirements Technical Debt (RTD) captures the consequences of sub-optimal decisions made concerning requirements, either deliberately (for strategic gains) or inadvertently (due to changes in context) ,during the identification, formalization, and implementation of requirements."

Examples:
- missing to capture important user needs
- introducing ambiguities when formalizing the requirements
- insufficient implementation of requirements



Practitioners may have particular needs regarding what they want to quantify when making informed decisions. They may want answers to particular questions such as the following when making their decisions for RTD management.

- How much VRTD has accumulated in the project right now?
- What is the Interest we are paying for not fixing the problems with veracity requirements?
- What is the benefit that can be gained by fixing problems with veracity requirements right now rather than later?
- What is the impact on the project due to changes in the veracity requirements?
- What are the consequences of VRTD for a given set of sub-optimally prioritized requirements?

PROGRESSION OF RESEARCH

Phases

Engagement with the NZ Organics Domain → **Systematic Literature Review** → **Online Survey on RTD and VRTD for software practitioners** → **Semi-structured Interviews with software practitioners**

Literature Review on Veracity attributes, Software Quality Models and dimensions of Culture

WORK IN PROGRESS

Register Here!

Key Findings / output

- The five types of veracity
- Classification of Veracity instances for Organic Certification based on types of veracity

Scan me!

Requirements Technical Debt Quantification Model (RTDQM)

- Cultural Veracity Requirements receive more attention in NZ compared to Rest of the world.
- Data Veracity Requirements are the most common type reported in the most critical incidents.
- Most respondents agree that the 'Impact on the Software Company' is high for veracitydebt.

References
1. https://veracity.wgtn.ac.nz/wp-content/uploads/2023/06/The-Veracity-Grand-Challenge-in-Computing_-A-Perspective-from-Aotearoa-New-Zealand.pdf
2. https://veracity.wgtn.ac.nz/wp-content/uploads/2024/03/8_Towards-quantifying-technical-debt-for-veracity-requirements.pdf
3. Perera, J., Tempero, E., Tu, Y.-C., & Blincoe, K. (2023). Quantifying Requirements Technical Debt: A Systematic Mapping Study and a Conceptual Model. 2023 IEEE 31st International Requirements Engineering Conference (RE), 123–133. doi:10.1109/RE57278.2023.00021
4. Li, Z., Avgeriou, P., & Liang, P. (2015). A systematic mapping study on technical debt and its management. Journal of Systems and Software, 101, 193–220. doi:10.1016/j.jss.2014.12.027
5. Alves, N. S. R., Mendes, T. S., de Mendonça, M. G., Spínola, R. O., Shull, F., & Seaman, C.(2016). Identification and management of technical debt: A systematic mapping study. Information and Software Technology, 70, 100–121. doi:10.1016/j.infsof.2015.10.008

National Science Challenges — SCIENCE FOR TECHNOLOGICAL INNOVATION — Kia kotahi mai – Te Ao Pūtaiao me Te Ao Hangarau

Victoria University of Wellington
University of Auckland University of Otago
University of Canterbury University of Waikato