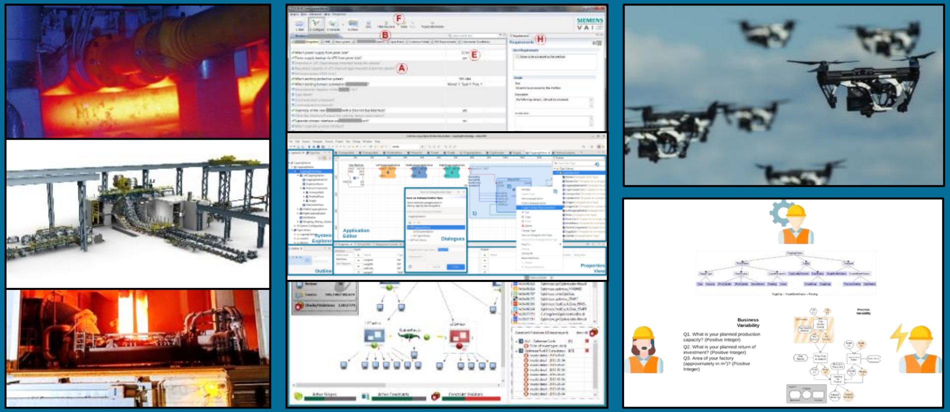
LINZ INSTITUTE
OF TECHNOLOGY

Software Engineering in Cyber-Physical Systems:
A Story of Variability and Complexity



Univ.-Prof. Mag. Dr. Rick Rabiser
LIT | Cyber-Physical Systems Lab
Johannes Kepler University Linz

Disclaimer: most of the things I present today have been done together with a team of people!

Rick Rabiser – rick.rabiser@jku.at
https://rickrabiser.github.io/rick/

1968 vs. 2018

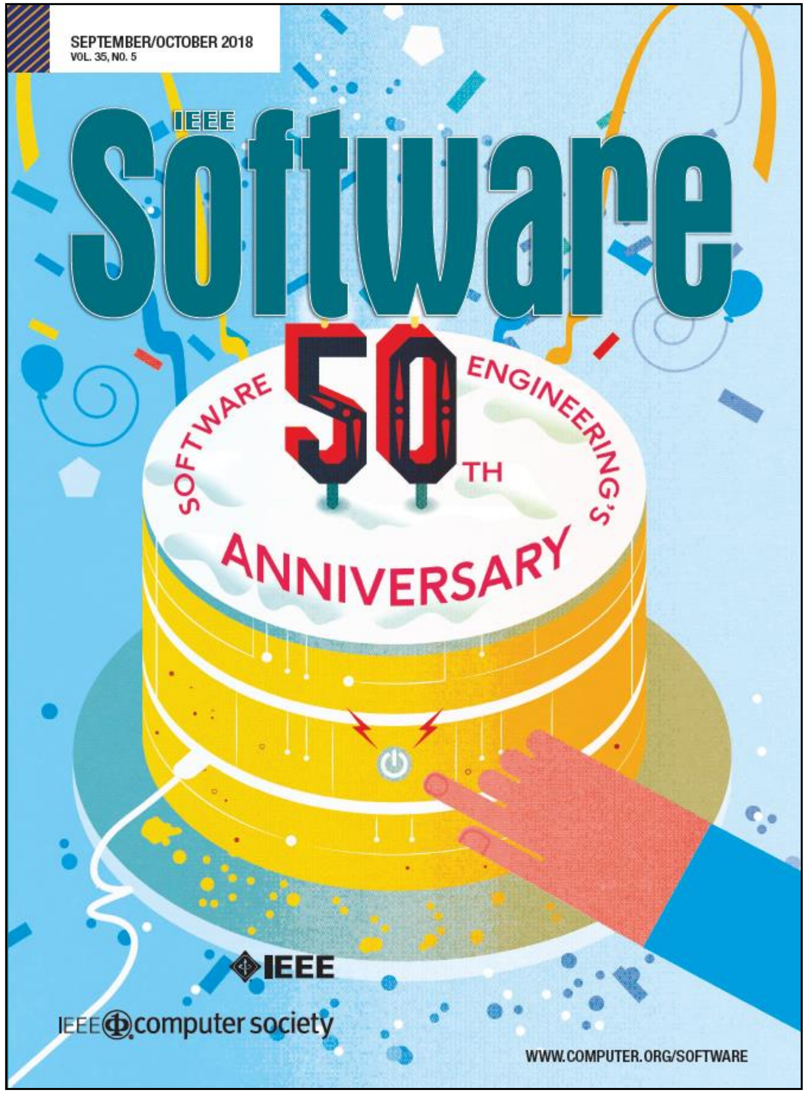
*We have to deal with complexity by **hierarchically ordering function and variability***

(E.W. Dijkstra: NATO Software Engineering Conference, 1968)

*We have to understand **how to design and implement reliable software** such that **maintenance**, **changeability**, and **variability** are easier to handle. […]*

***Software engineering is** part of systems engineering of **cyber-physical systems** closely connected to the real world.*

(M. Broy: IEEE Software, 2018)

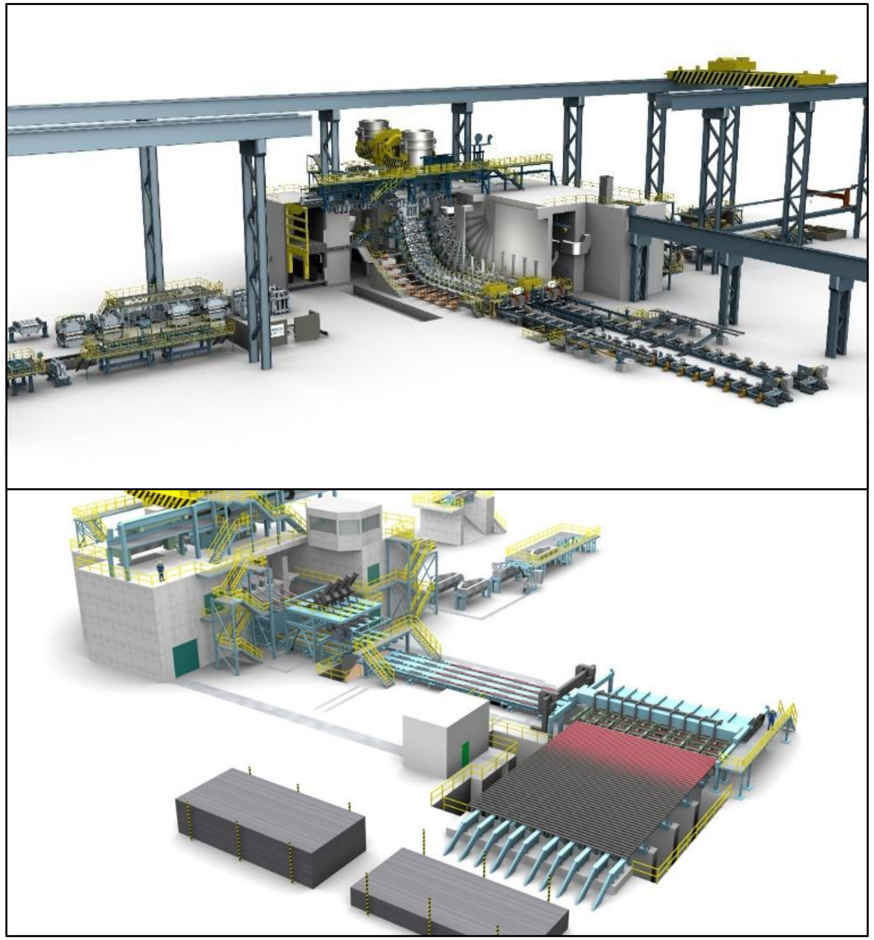


Cyber-Physical Systems I've worked with (just a selection)…

SIEMENS

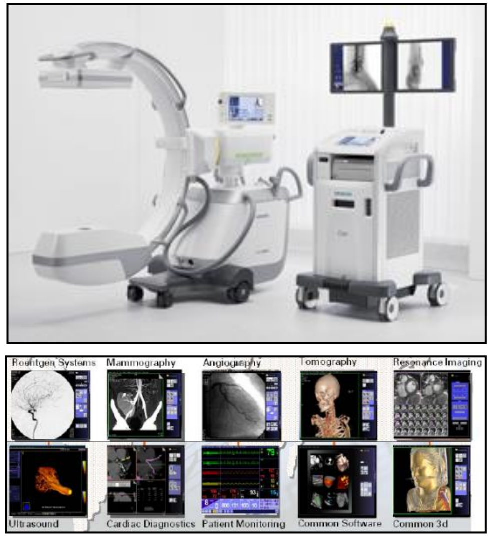
PRIMETALS TECHNOLOGIES

Continuous Casting Machines

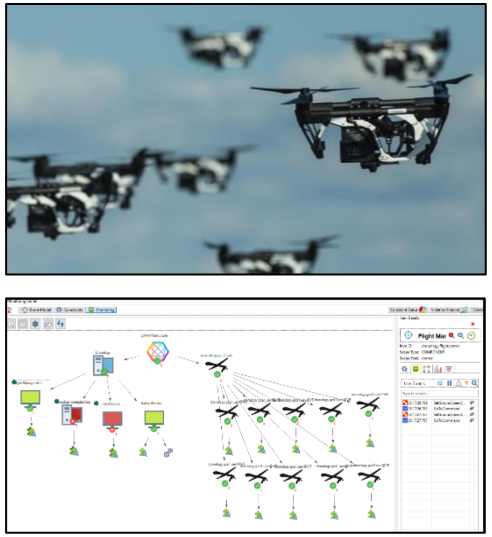


Electric Arc Furnaces

Medical Systems



Drone Control Systems

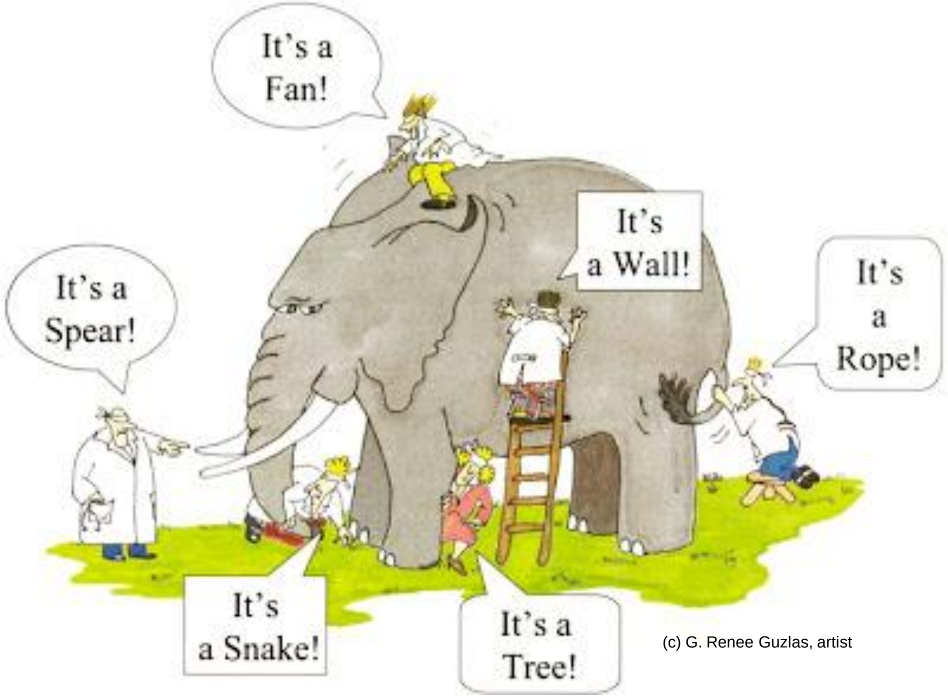


Blast Furnaces

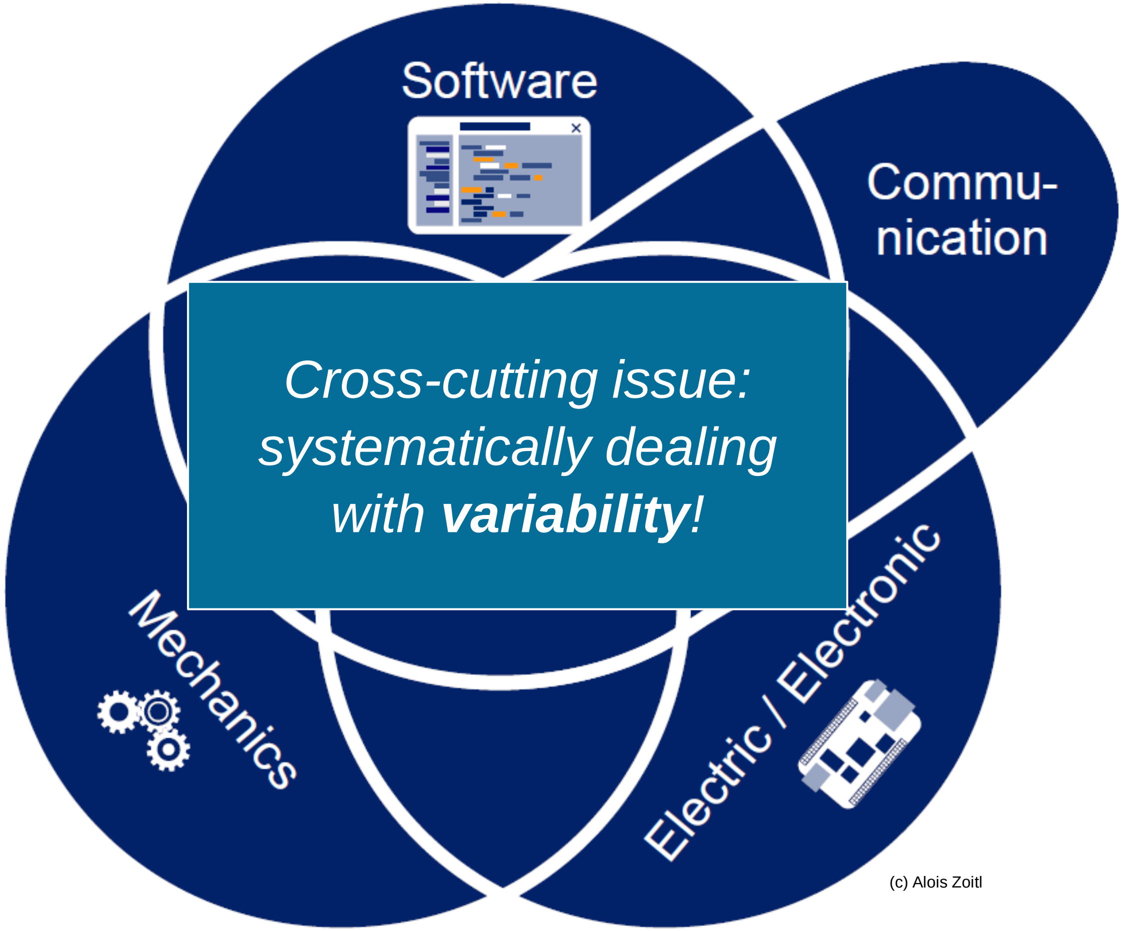
Rolling Mills

Rick Rabiser – rick.rabiser@jku.at
https://rickrabiser.github.io/rick/

Key Challenge of working with CPS: Multidisciplinarity



(c) G. Renee Guzlas, artist

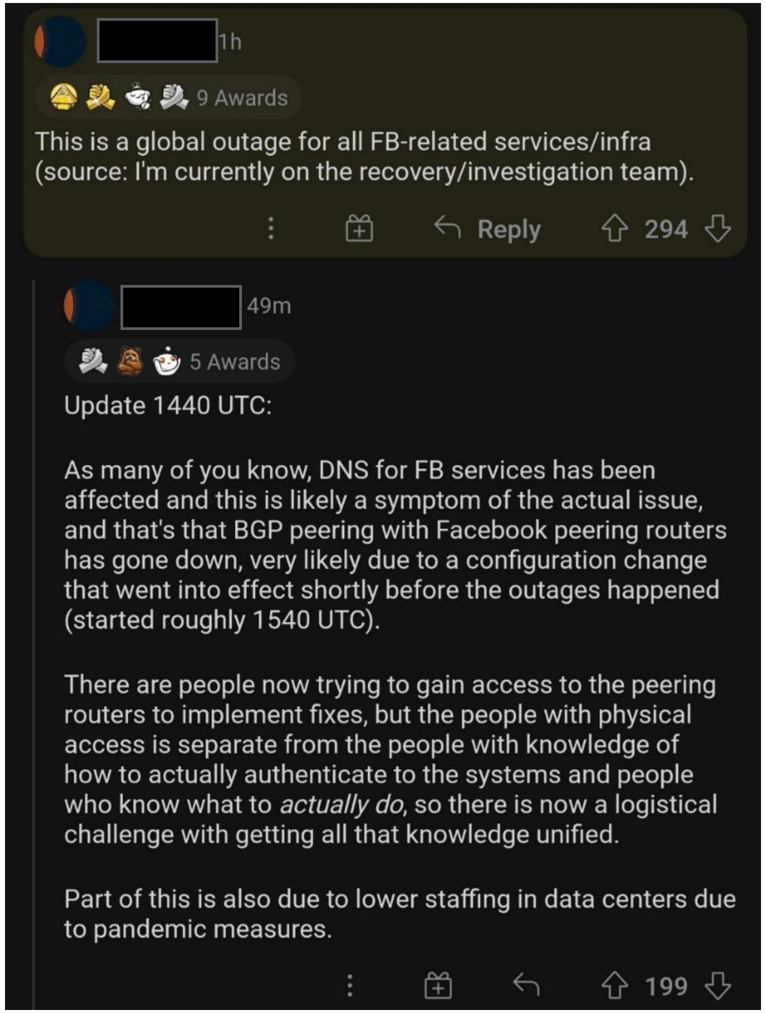


*Cross-cutting issue: systematically dealing with **variability**!*

(c) Alois Zoitl

Variability and Multidisciplinarity in Practice

- Oct 4/5, 2021: Facebook, Instagram, WhatsApp Down for several hours (7h+) world-wide
- Facebook Stocks down by more than 5 percent, over 40 Billion US Dollars
- Reason? Software configuration error with effects on routing!
- Why so long to fix problem?
  - People with knowledge about routers are not the same people with knowledge how to fix the software configuration problem



https://en.wikipedia.org/wiki/2021_Facebook_outage

Industry and Variability


FOCUS: GUEST EDITORS' INTRODUCTION

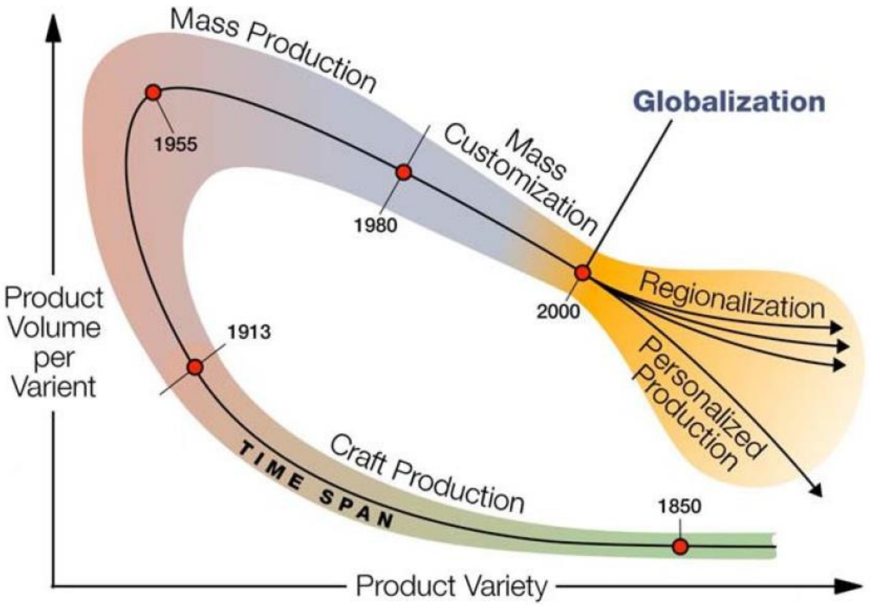
TRENDS IN SYSTEMS AND SOFTWARE VARIABILITY

Jan Bosch, Chalmers University of Technology

Rafael Capilla, Rey Juan Carlos University

Rich Hilliard, consulting software systems architect

"Industrial reality shows the number of variation points, variants, and dependencies easily reaches staggering levels [...] often resulting in a situation in which **no one** […] **has a comprehensive overview of the available variability**."



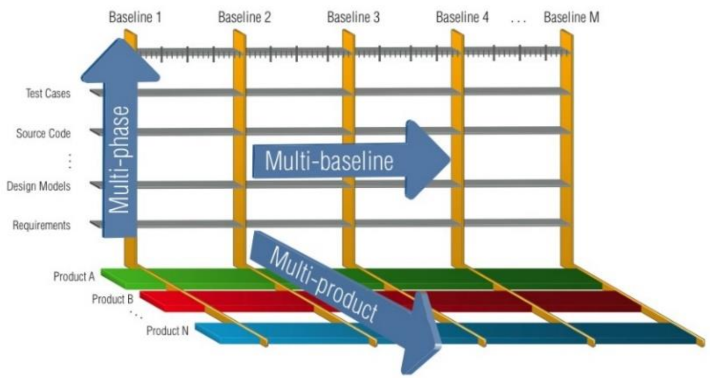
Source: Yoram Koren, The Global Manufacturing Revolution, 2010

Jan Bosch, Raffael Capilla, Rich Hilliard: Trends in Systems and Software Variability. IEEE Software, vol. 32(3), pp. 44-51, 2015.

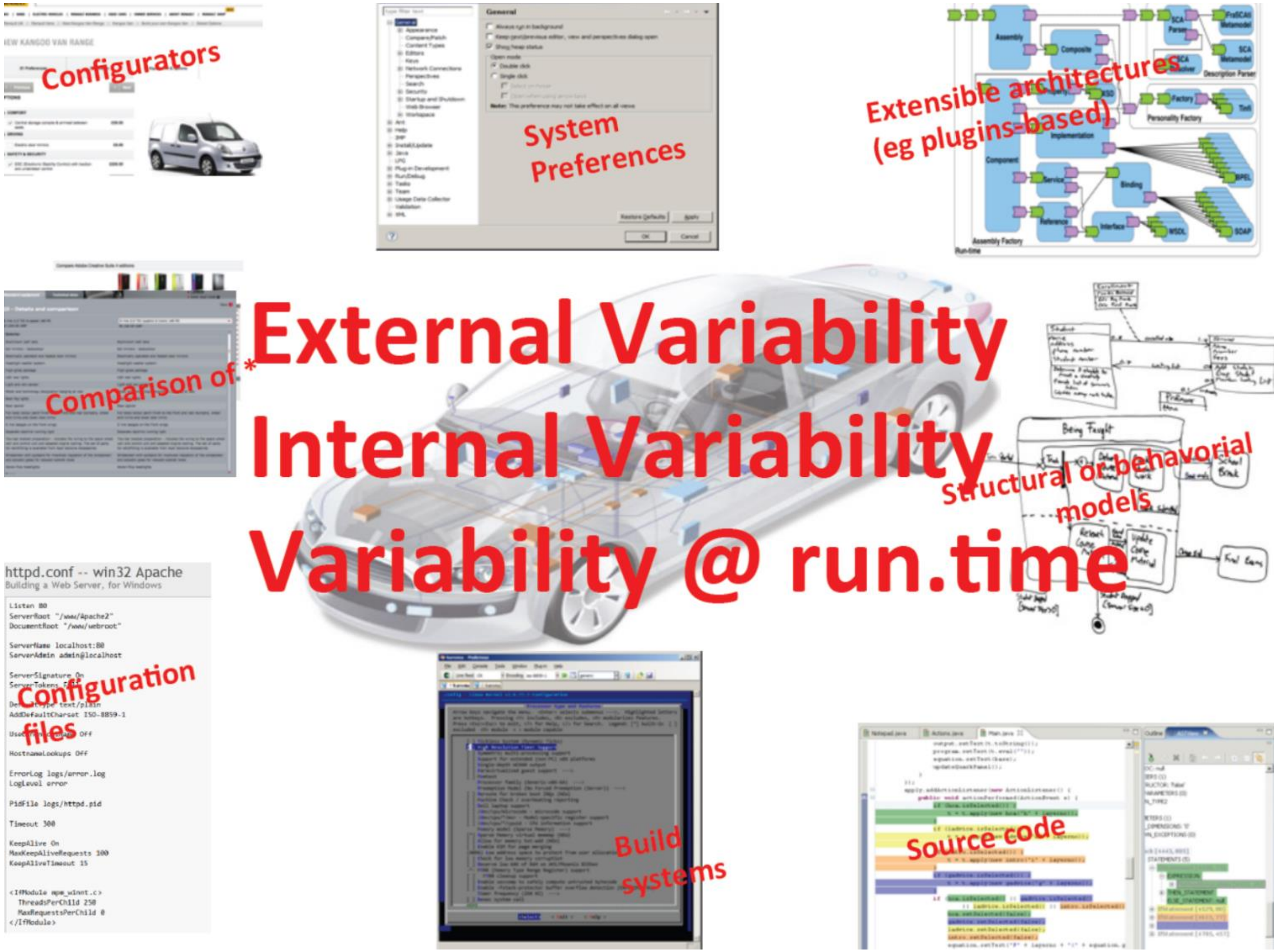
Variability?

“the ability of a system or artifact to be efficiently extended, changed, customized or configured for use in a particular context”

(Capilla et al. 2013)



(c) Charles Krueger, BigLever

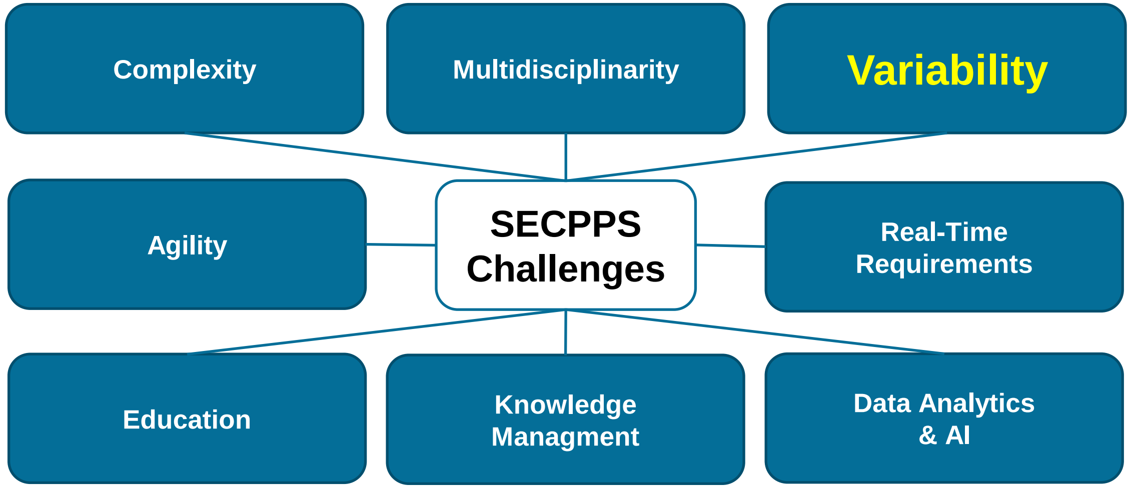


ETFA 2022
Industry Voices

ETFA 2023
DSLs Maturity Model

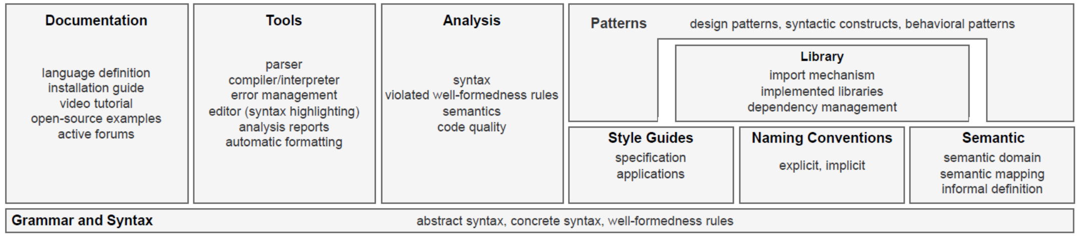
SECPPS (Software Engineering in Cyber-Physical Production Systems) and Variability

- **Goal**: address the challenges in adopting state-of-the-art SE tools and techniques in the production automation domain and discuss various approaches to tackle the issues
- **Founded** by: Rick Rabiser, Birgit Vogel-Heuser (TU Munich), Manuel Wimmer (WIN/SE, JKU), Andreas Wortmann (Uni Stuttgart), Alois Zoitl (JKU)



Language	GS	Sem	Nam	Sty	Lib	Pat	A	T	D
IEC 61499	◓	●	◓	○	◓	●	◓	◓	◓
SysML v1.6	◓	◓	◓	◓	●	◓	○	◓	●
MontiArc	●	●	●	◓	●	○	◓	◓	●
IVML	◓	●	○	○	◓	◓	◓	◓	◓
PPRDSL	◓	●	◓	○	◓	○	◓	◓	◓

DSL Ecosystem Maturity Model



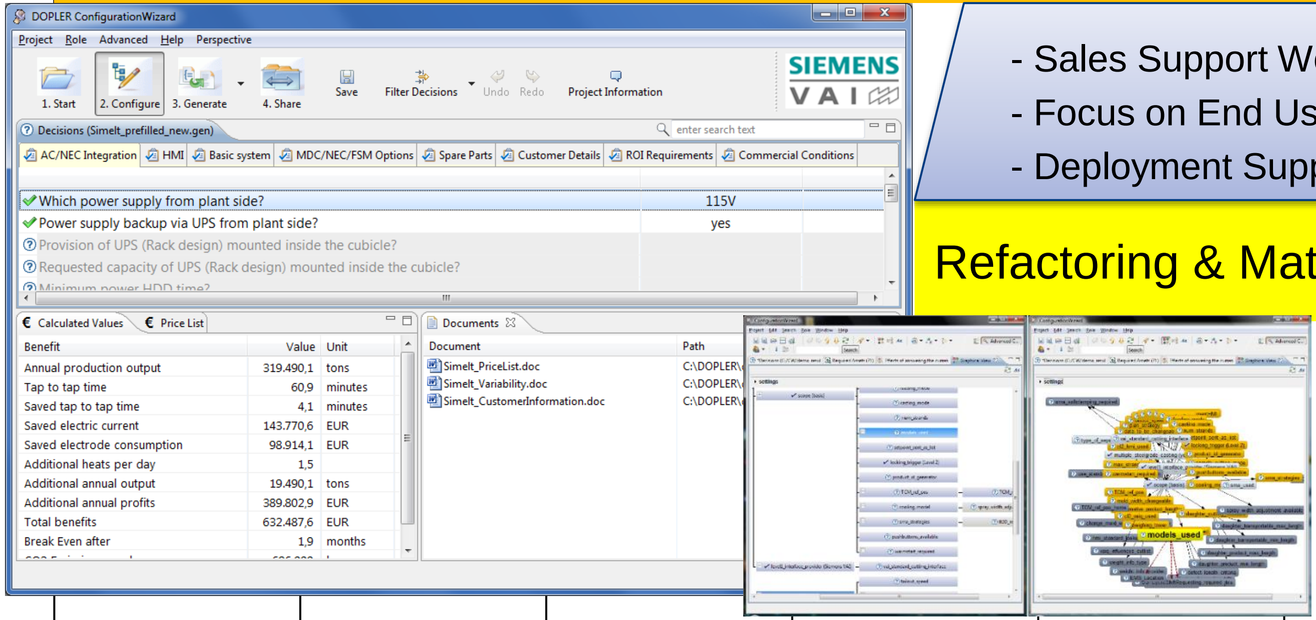
Earlier Research: My Journey of Developing a Configuration Tool 2006-2013

- Flexible Variability Modeling Approach
- Derivation Model
- Config. Prototype

- Model Fragments
- Evolution Support
- Case Studies

- Multi Product Lines
- Product Line Bundles
- Distributed Configuration

Research Results from Christian Doppler Lab ASE



- Sales Support Workflow
- Focus on End Users
- Deployment Support

Refactoring & Maturation Project

- Conf. Tool for Generating Documentation / Business Calculations / etc.

Industrial Rollout at SVAI

2006 2007 2008 2009 2010 2011 2012 2013

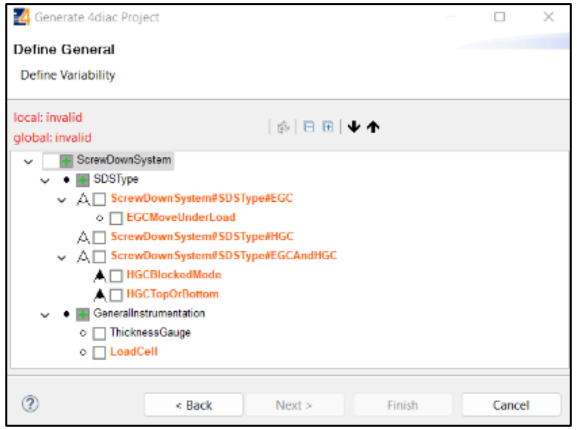
Recent Research: V4rdiac: Multidisciplinary Delta-Oriented Variability Management in CPPS

Expressing variability from different aspects (e.g., business, signal, process) using heterogeneus **variability models**

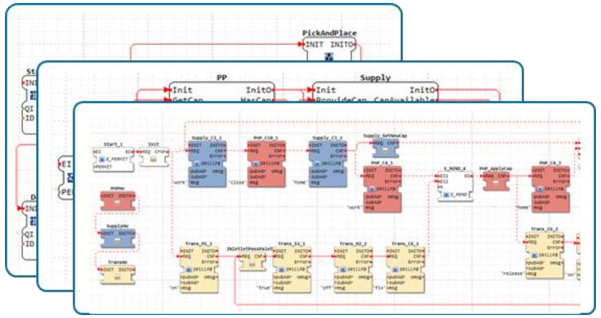
```
delta DApplyCapPresNotReached;
uses Application;

{
    <Modify> Subapplication PnP_ApplyCap {
        <Add> EventOutput Error type=Event;
        <Add> FB PnP_PressureSens_3 type=SkillFB;
        <Add> EventConnection PnP_C5_1.CNF
            PnP_PressureSens_3.REQ;
        <Add> EventConnection PnP_PressureSens_3.CNF
            PnP_PressureSens_2.REQ;
        <Remove> EventConnection PnP_C5_1.CNF
            PnP_PressureSens_2.REQ;
    }
}
```

Express control software variability using **delta models**



Product configuration interface based on variability models

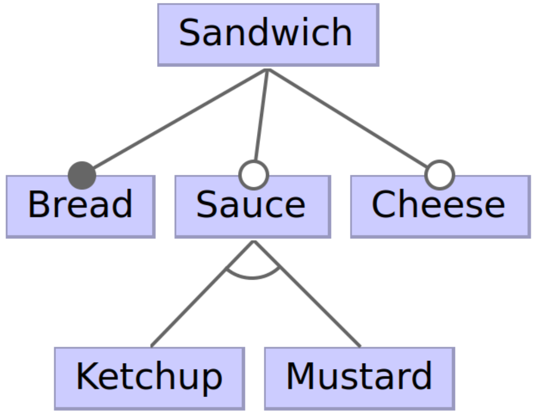


Control software generator based on selected configuration options

Recent Research: Universal Variability Language (UVL)

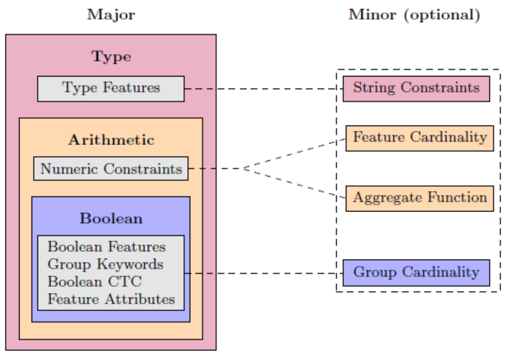
- A community effort towards a unified language for variability models
  - https://universal-variability-language.github.io/
- UVL is a direct result of the efforts within the initiative
  - https://modevar.github.io/
- Want to try it? https://uvl.uni-ulm.de/

```
features
    Sandwich
        mandatory
            Bread
        optional
            Sauce
                alternative
                    Ketchup
                    Mustard
            Cheese
constraints
    Ketchup => Cheese
```



UVL Support

- Language Levels (Boolean, Arithmetic, Type, …)
- Multi modeling concept (via imports)



- UVL **Tool Support**
  - (ANTLR) Parser Implementations for Python and Java
  - Integrated in FeatureIDE and FLAMA and TRAVART
  - Importers/exporters for other tools, e.g., pure::variants
  - Rust-based Language Server Protocol for integrating UVL in any IDE: UVLS
  - Web-based playground based on UVLS UVL Playground
  - Visual Studio Code extension based on UVLS

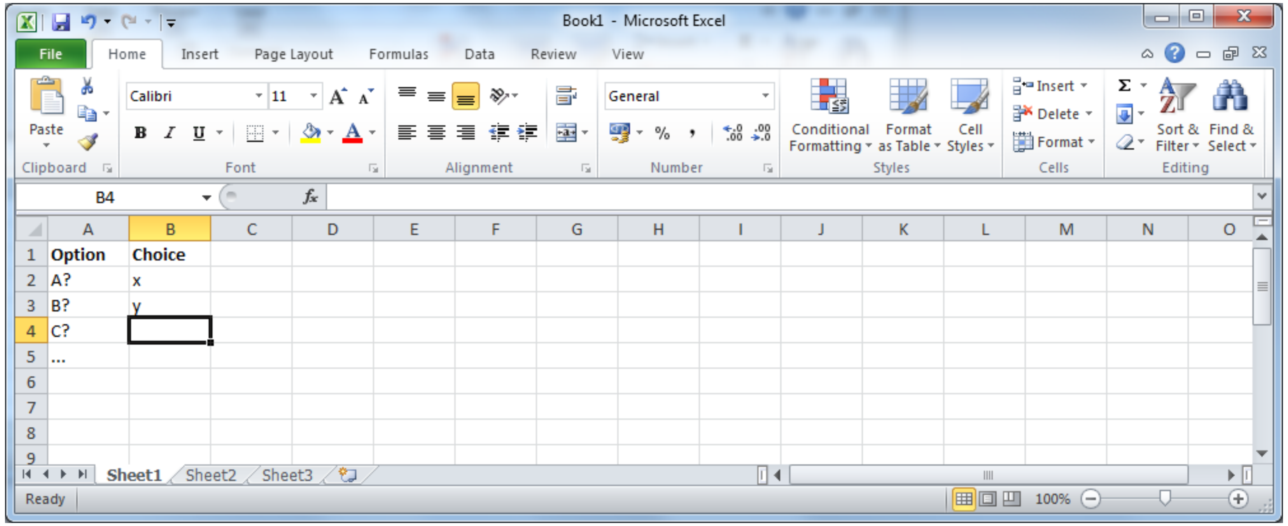
- Examples: https://github.com/Universal-Variability-Language/uvl-models

UVL Publications

- Chico Sundermann, Kevin Feichtinger, Dominik Engelhardt, Rick Rabiser, and Thomas Thüm. 2021. Yet another textual variability language? a community effort towards a unified language. SPLC '21. https://doi.org/10.1145/3461001.3471145
- Chico Sundermann, Stefan Vill, Thomas Thüm, Kevin Feichtinger, Prankur Agarwal, Rick Rabiser, José A. Galindo, and David Benavides. 2023. UVLParser: Extending UVL with Language Levels and Conversion Strategies. SPLC '23 - Tool Track. https://doi.org/10.1145/3579028.3609013
- Chico Sundermann, Tobias Heß, Dominik Engelhardt, Rahel Arens, Johannes Herschel, Kevin Jedelhauser, Benedikt Jutz, Sebastian Krieter, and Ina Schaefer. 2021. Integration of UVL in FeatureIDE. MODEVAR@SPLC '21. https://doi.org/10.1145/3461002.3473940
- Kevin Feichtinger, Johann Stöbich, Dario Romano, and Rick Rabiser. 2021. TRAVART: An Approach for Transforming Variability Models. VaMoS '21. https://doi.org/10.1145/3442391.3442400
- Dario Romano, Kevin Feichtinger, Danilo Beuche, Uwe Ryssel, and Rick Rabiser. 2022. Bridging the gap between academia and industry: transforming the universal variability language to pure::variants and back. MODEVAR@SPLC '22. https://doi.org/10.1145/3503229.3547056
- José A. Galindo and David Benavides. 2020. A Python framework for the automated analysis of feature models: A first step to integrate community efforts. MODEVAR@SPLC '20. https://doi.org/10.1145/3382026.3425773José A. Galindo, Jose-Miguel Horcas, Alexander Felferning, David Fernandez-Amoros, and David Benavides. 2023. FLAMA: A collaborative effort to build a new framework for the automated analysis of feature models. SPLC' 23 - Tool Track. https://doi.org/10.1145/3579028.3609008
- Jose M. Horcas, Jose A. Galindo, Mónica Pinto, Lidia Fuentes, and David Benavides. 2022. FM fact label: a configurable and interactive visualization of feature model characterizations. SPLC '22. https://doi.org/10.1145/3503229.3547025
- David Romero, José Á. Galindo, Jose-Miguel Horcas, and David Benavides. 2021. A first prototype of a new repository for feature model exchange and knowledge sharing. SPLC '21. https://doi.org/10.1145/3461002.3473949

Tools? THE most used SPL/Variability Mgmt Tool?

- Microsoft Excel!



„The Williams car build workbook, with roughly 20,000 individual parts”

/. Slashdot | Stories | Firehose › | All | Popular | Polls | /. Software | Newsletter | Jobs | Submit | Search

Topics: Devices Build Entertainment Technology Open Source Science YRO

Formula 1 Chief Appalled To Find Team Using Excel To Manage 20,000 Car Parts (arstechnica.com)

57

Posted by BeauHD on Wednesday March 20, 2024 @11:30PM from the Excel-blame-game dept.

An anonymous reader quotes a report from Ars Technica:

> Starting in early 2023, Williams team principal James Vowles and chief technical officer Pat Fry started reworking the F1 team's systems for designing and building its car. It would be painful, but the pain would keep the team from falling even further behind. As they started figuring out new processes and systems, they encountered what they considered a core issue: Microsoft Excel. The Williams car build workbook, with roughly 20,000 individual parts, was "a joke," Vowles recently told The Race. "Impossible to navigate and impossible to update." This colossal Excel file lacked information on how much each of those parts cost and the time it took to produce them, along with whether the parts were already on order. Prioritizing one car section over another, from manufacture through inspection, was impossible, Vowles suggested.
>
> "When you start tracking now hundreds of thousands of components through your organization moving around, an Excel spreadsheet is useless," Vowles told The Race. Because of the multiple states each part could be in -- ordered, backordered, inspected, returned -- humans are often left to work out the details. "And once you start putting that level of complexity in, which is where modern Formula 1 is, the Excel spreadsheet falls over, and humans fall over. And that's exactly where we are." The consequences of this row/column chaos, and the resulting hiccups, were many. Williams missed early pre-season testing in 2019. Workers sometimes had to physically search the team's factory for parts. The wrong parts got priority, other parts came late, and some piled up. And yet transitioning to a modern tracking system was "viciously expensive," Fry told The Race, and making up for the painful process required "humans pushing themselves to the absolute limits and breaking."
>
> The idea that a modern Formula 1 team, building some of the most fantastically advanced and efficient machines on Earth, would be using Excel to build those machines might strike you as odd. F1 cars cost an estimated $12-$16 million each, with resource cap of about $145 million. But none of this really matters, and it actually makes sense, if you've ever worked IT at nearly any decent-sized organization. Then again, it's not even uncommon in Formula 1. When Sebastian Anthony embedded with the Renault team, he reported back for Ars in 2017 that Renault Sport Formula One's Excel design and build spreadsheet was 77,000 lines long -- more than three times as large as the Williams setup that spurred an internal revolution in 2023.
>
> Every F1 team has its own software setup, Anthony wrote, but they have to integrate with a lot of other systems: Computational Fluid Dynamics (CFD) and wind tunnel results, rapid prototyping and manufacturing, and inventory. This leaves F1 teams "susceptible to the plague of legacy software," Anthony wrote, though he noted that Renault had moved on to a more dynamic cloud-based system that year. (Renault was also "a big Microsoft shop" in other areas, like email and file sharing, at the time.) One year prior to Anthony's excavation, Adam Banks wrote for Ars about the benefits of adopting cloud-based tools for enterprise resource planning (ERP). You adopt a cloud-based business management software to go "Beyond Excel." "If PowerPoint is the universal language businesses use to talk to one another, their internal monologue is Excel," Banks wrote. The issue is that all the systems and processes a business touches are complex and generate all kinds of data, but Excel is totally cool with taking in all of it. Or at least 1,048,576 rows of it. Banks cited Tim Worstall's 2013 contention that Excel could be "the most dangerous software on the planet." Back then, international investment bankers were found manually copying and pasting Excel between Excel sheets to do their work, and it raised alarm.

https://tech.slashdot.org/story/24/03/20/2142229/formula-1-chief-appalled-to-find-team-using-excel-to-manage-20000-car-parts

Custom-developed „PL“ Tools

- Typical scenario:
  - Company recognizes need to manage variability
  - Company implements their own solution to manage variability

- However, there are **many commercial and academic PL tools** existing
  - pure::variants (now part of PTC), FeatureIDE (Open Source FOSD IDE), …

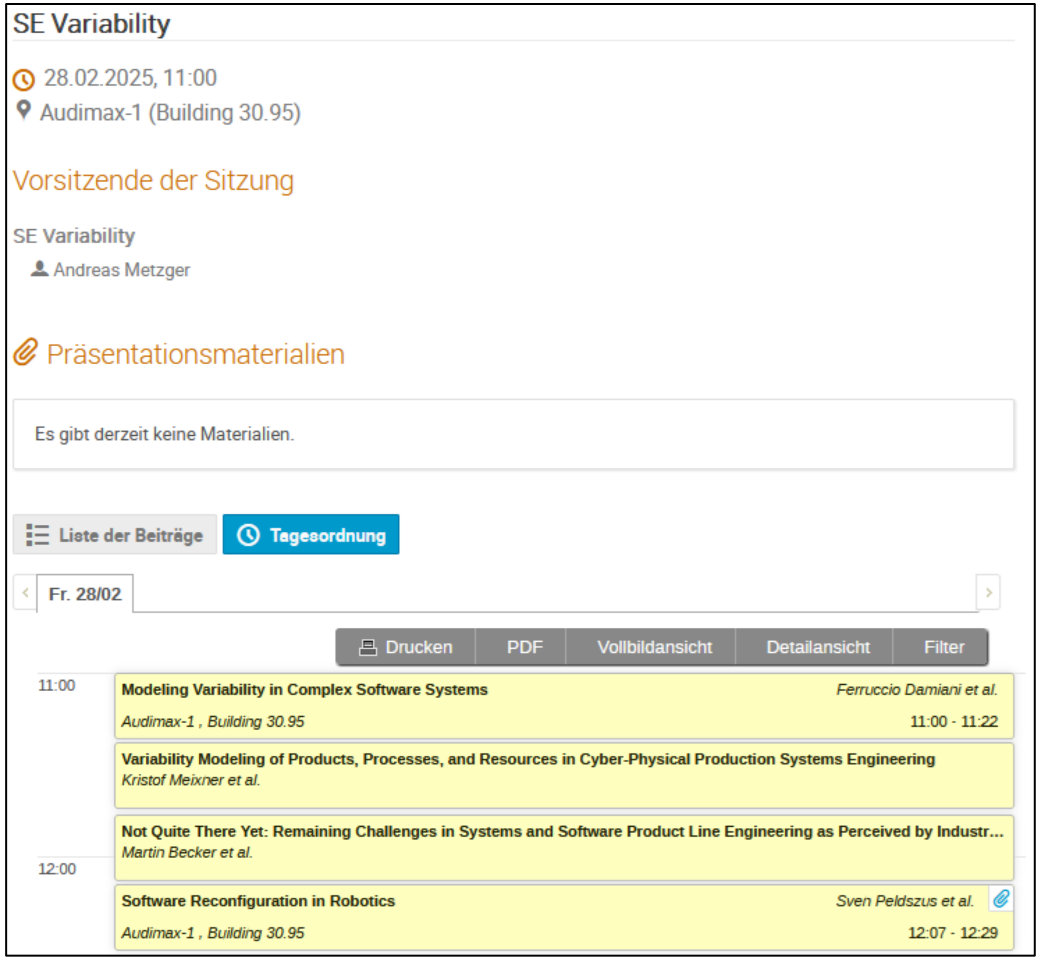
Recent Work: Remaining Industry Challenges?

Case	Domain	Focus	Main Challenges
1	Automotive mechatronics	Broaden product portfolio	Standards, "docs as code", CI/CD, test strategy, project-based org.
2	Sensors, measuring devices	Variant management & product configuration	PL verification
3	HVAC, home applicances	System of systems PLE	Ecosystem, multiple domains, verification and validation effort for variants, portfolio vs. engineering across distributed locations
4	Metallurgical plant solutions	Systematic variability management	Dependence on automation platform vendors, variability on multiple levels and in multiple disciplines
5	Agritech	Multi systems PL	Long-living systems, small production volumes, multiple product lines, maintenance, tools for modeling and simulation
6	Rail transport, rolling stock	Enhance reuse rate	Documenting, scoping and evaluating PL architectures, var. modeling, standardising modules, module maintenance, org. structures
7	Industrial automation	New generation of automation products	Knowledge silos, integration and testing, HIL testing, perception of slow platform development, modification of shared assets
8	Defense, aerospace	Increase modularisation and reuse	Perception of PLE, design authority, asynchronous information, governance, reuse scope, proactive reuse identification, frequent analysis
9	Automotive powertrain controllers	PL variant management	Consistent variant management across disciplines, variant management in V & V, evolution of PLs, PL of PLs, efficient var. realisation, usability of var. management tools, collaboration with OEMs

M. Becker, R. Rabiser, G. Botterweck: Not Quite There Yet: Remaining Challenges in Systems and Software Product Line Engineering as Perceived by Industry Practitioners, SPLC 2024.

Come to my SE Talk

- Fr, 11:45, Audimax-1
- Not Quite There Yet: Remaining Challenges in Systems and Software PLE as Perceived by Industry
- Or read the paper



Rick Rabiser – rick.rabiser@jku.at
https://rickrabiser.github.io/rick/

Useful Material/Further Reading: Useful Links

- Variability Modeling Body of Knowledge: https://github.com/SECPS/VMBoK
- Universal Variability Language: https://universal-variability-language.github.io/
- MODEVAR Initiative: https://modevar.github.io/
- Open Source Feature-oriented SW Development IDE: https://featureide.github.io/
- Online Var Modeling and Configuration Tool: https://variability.dev/
- Software Product Line Conference: https://splc.net/
- Variability Modeling Conference: https://vamosconf.net/
- Repository of Teaching Material: http://teaching.variability.io/
- Repository of Case Studies: https://but4reuse.github.io/espla_catalog/

Useful Material/Further Reading: Textbooks

- S. Apel, D. Batory, C. Kästner, and G. Saake, Feature-Oriented Software Development: Concepts and Implementation: Springer, 2013.
- R. Capilla, J. Bosch, and K. Kang, Systems and Software Variability Management: Concepts, Tools and Experiences: Springer, 2013.
- P. Clements and L. Northrop, Software Product Lines: Practices and Patterns: SEI Series in Software Engineering, Addison-Wesley, 2001.
- J. Martinez, W. K. G. Assunção, T. Ziadi, M. Acher, S. Vergilio, S., Handbook of Re-Engineering Software Intensive Systems into Software Product Lines. R. E. Lopez-Herrejon (Ed.). Springer 2023.
- K. Pohl, G. Böckle, and F. van der Linden, Software Product Line Engineering: Foundations, Principles, and Techniques: Springer, 2005.
- F. van der Linden, K. Schmid, and E. Rommes, Software Product Lines in Action - The Best Industrial Practice in Product Line Engineering: Springer Berlin Heidelberg, 2007.

Useful Material/Further Reading: Papers (to start with and find other material)

- M. Becker, R. Rabiser, G. Botterweck: Not Quite There Yet: Remaining Challenges in Systems and Software Product Line Engineering as Perceived by Industry Practitioners, SPLC 2024.
- T. Berger, J.P. Steghöfer, T. Ziadi, J. Robin, J. Martinez, J.: The state of adoption and the challenges of systematic variability management in industry. Empirical Software Engineering, 25, 1755-1797, 2020.
- K. Czarnecki, P. Grünbacher, R. Rabiser, K. Schmid, and A. Wasowski: Cool Features and Tough Decisions: A Comparison of Variability Modeling Approaches. VaMoS 2012.
- M. Galster, D. Weyns, D. Tofan, B. Michalik, P. Avgeriou: Variability in software systems—a systematic literature review. IEEE Transactions on Software Engineering, 40(3), 282-306, 2013.
- M. Raatikainen, J. Tiihonen, and T. Männistö: Software product lines and variability modeling: A tertiary study. Journal of Systems and Software, vol. 149, pp. 485-510, 2019.
- R. Rabiser, K. Schmid, M. Becker, G. Botterweck, M. Galster, I. Groher, D. Weyns: A Study and Comparison of Industrial vs. Academic Software Product Line Research Published at SPLC, SPLC 2018.

Variability Model(s) (Repositories)

- https://github.com/SoftVarE-Group/feature-model-benchmark/tree/master/feature_models
- https://www.uvlhub.io/
- https://but4reuse.github.io/espla_catalog/
- https://github.com/tuw-qse/cpps-var-case-studies

Thank you!

Rick Rabiser | rick.rabiser@jku.at
Christian Doppler Lab VaSiCS
LIT | Cyber-Physical Systems Lab
Johannes Kepler University Linz

JKU LINZ INSTITUTE OF TECHNOLOGY
PRIMETALS TECHNOLOGIES
oberösterreich
Federal Ministry Republic of Austria Digital and Economic Affairs