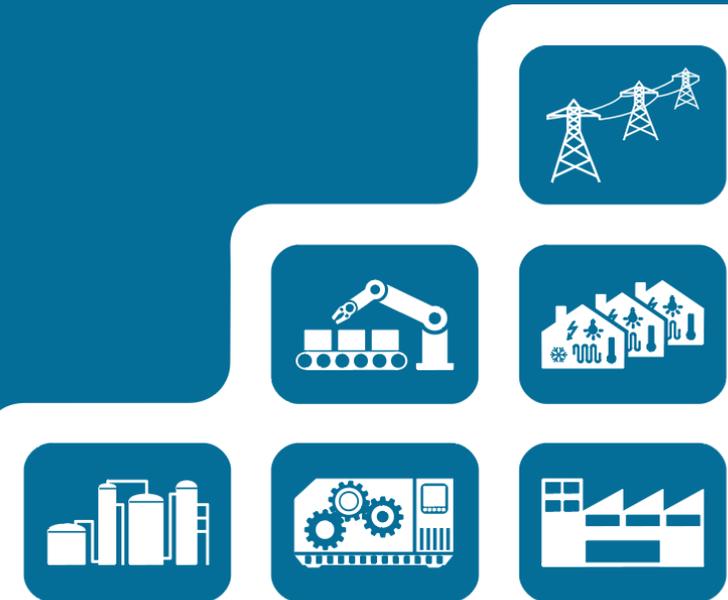


Thesis Topics (and PR) at the LIT CPS Lab (and the CD Lab VaSiCS)



Univ. Prof. Dr. Rick Rabiser and Univ.-Prof. Dr. Alois Zoitl
Christian Doppler Lab VaSiCS
LIT | Cyber-Physical Systems Lab
Johannes Kepler University Linz

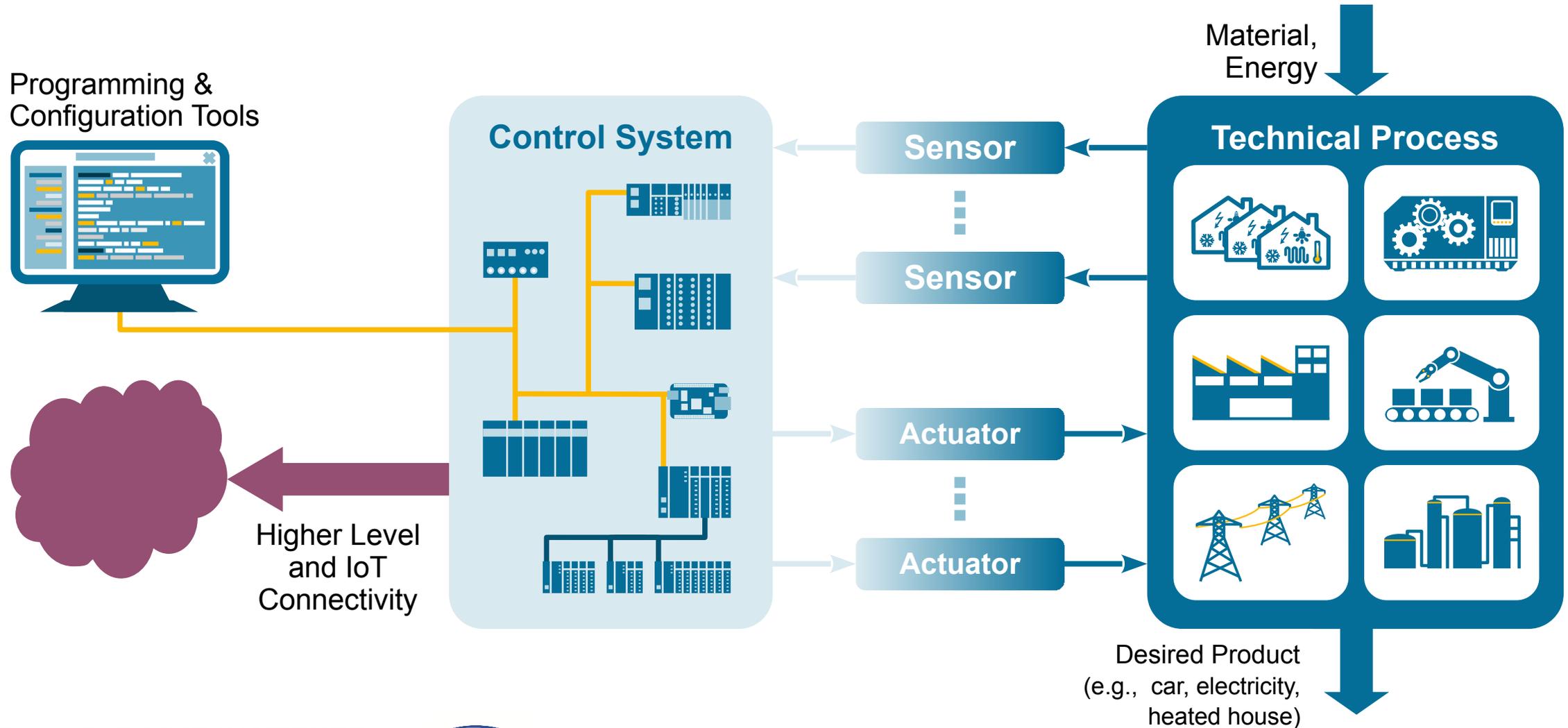


LIT Cyber-Physical Systems Lab

<https://www.jku.at/lit/cps-lab>



Background Cyber-Physical Systems

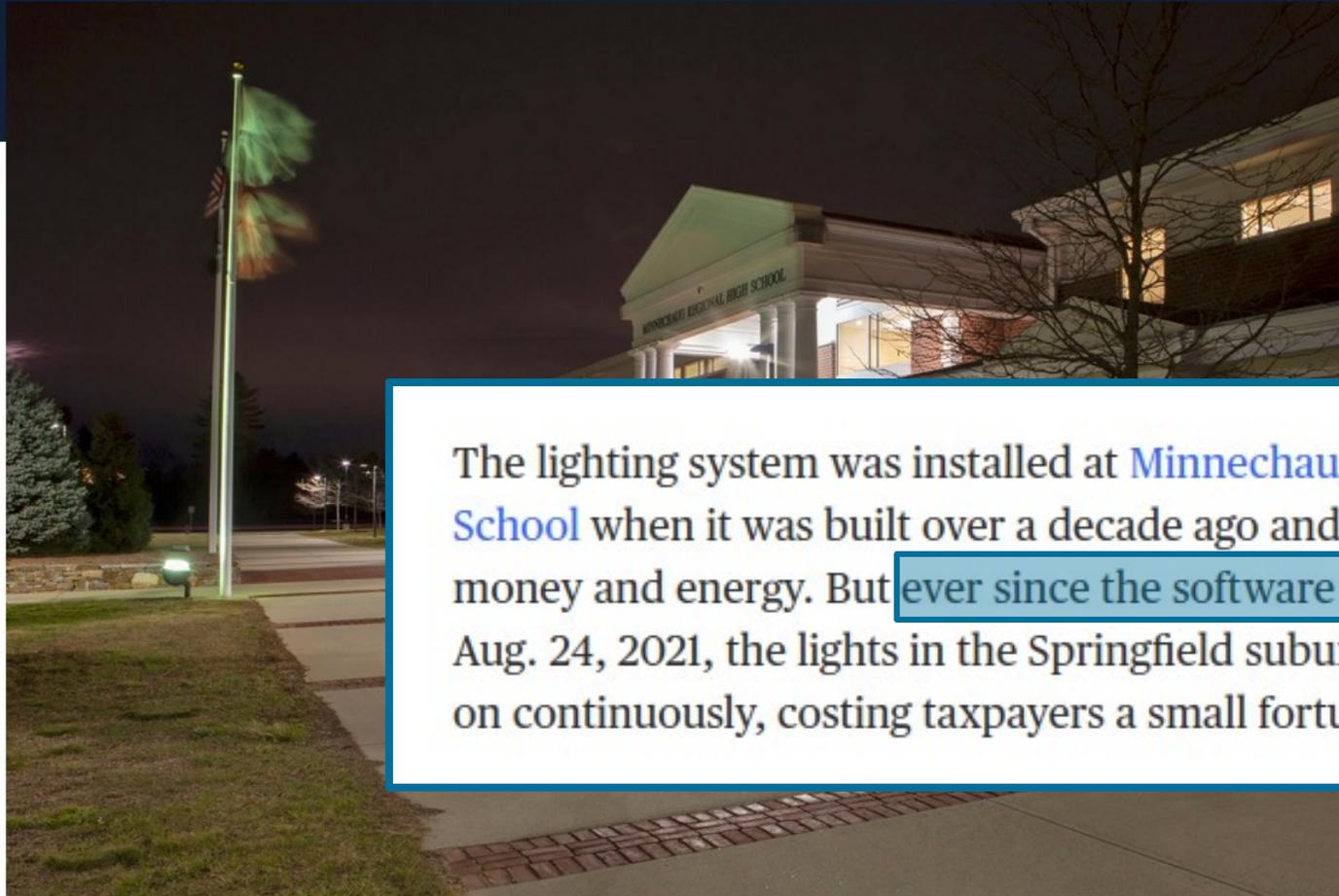


EXCLUSIVE

U.S. NEWS

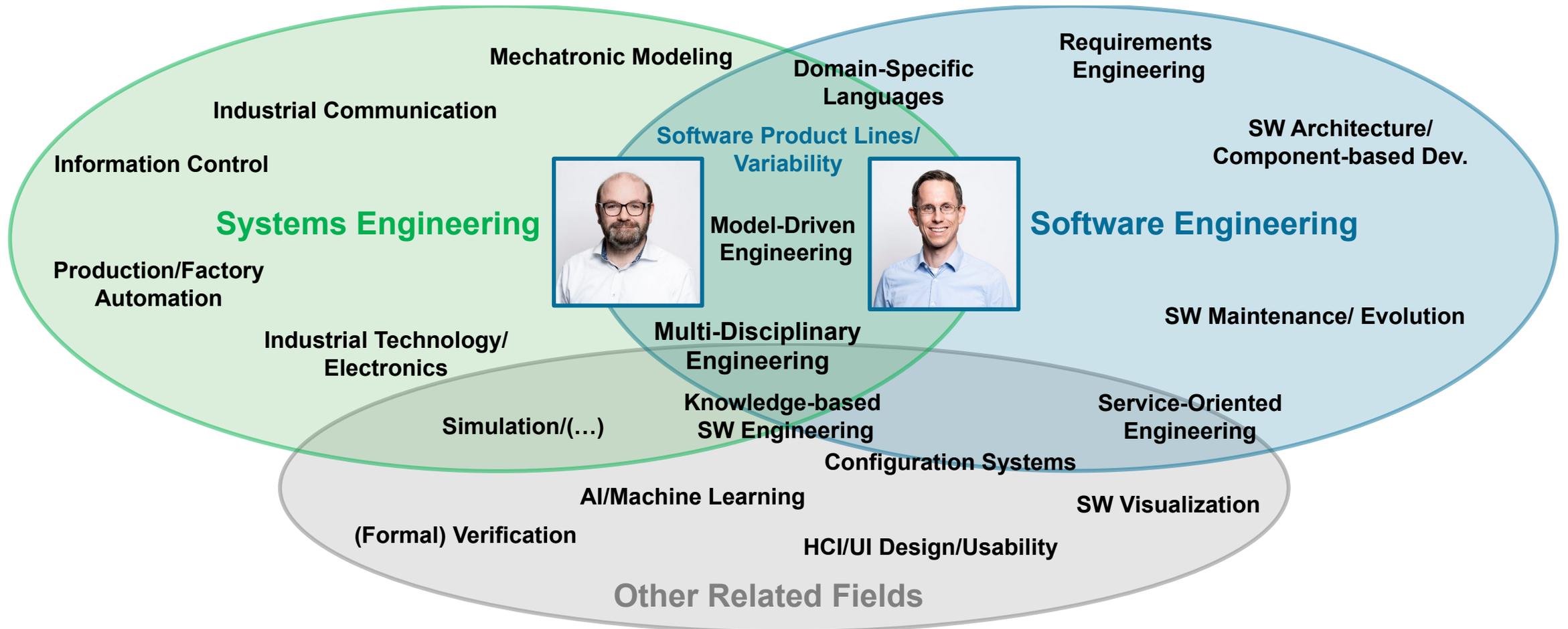
The lights have been on at a Massachusetts school for over a year because no one can turn them off

Blame it on the pandemic and "supply chain problems," says the school district's assistant superintendent of finance.



The lighting system was installed at [Minnechaug Regional High School](#) when it was built over a decade ago and was intended to save money and energy. But **ever since the software that runs it failed** on Aug. 24, 2021, the lights in the Springfield suburbs school have been on continuously, costing taxpayers a small fortune.

Scientific Landscape and LIT CPS Lab Professors





Selected Current Projects & Possible Topics

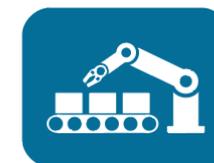
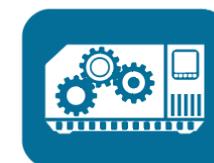


Please note: for all these projects, practica and theses are possible (Bachelor, Masters, PhD)

Christian Doppler Lab VaSiCS

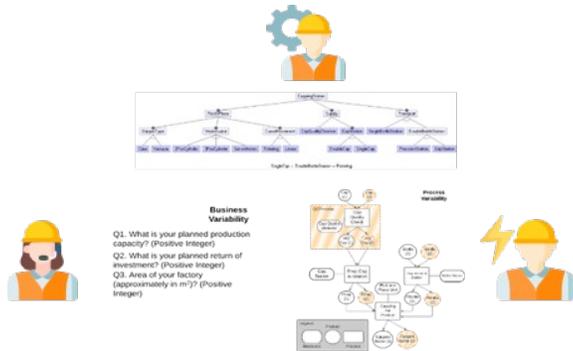
LIT | Cyber-Physical Systems Lab

Johannes Kepler University Linz





Multidisciplinary Delta-Oriented Variability Management in Cyber-Physical Systems



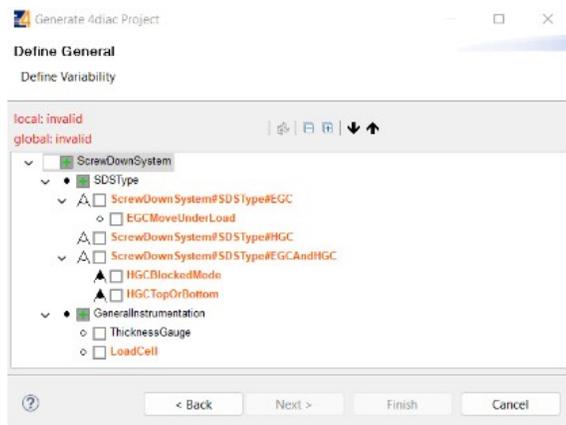
Expressing variability from different aspects (e.g., business, signal, process) using multiple variability models

```

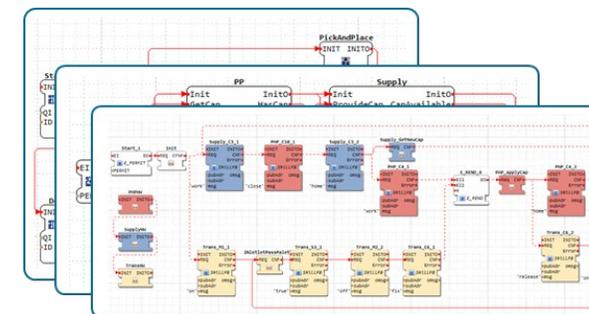
delta DApplyCapPresNotReached;
uses Application;

{
  <Modify> Subapplication PnP_ApplyCap {
    <Add> EventOutput Error type=Event;
    <Add> FB PnP_PressureSens_3 type=SkillIFB;
    <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

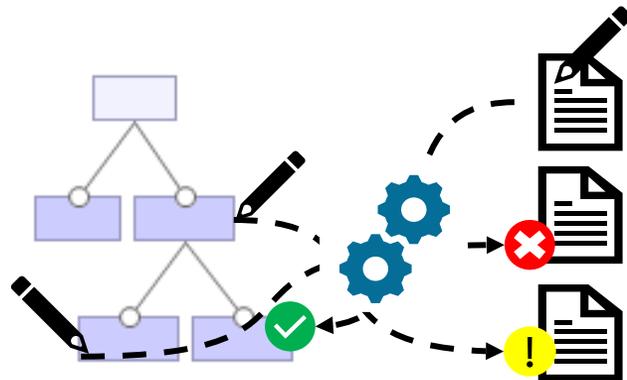


Control software generator based on selected configuration options



Examples for Concrete Topics

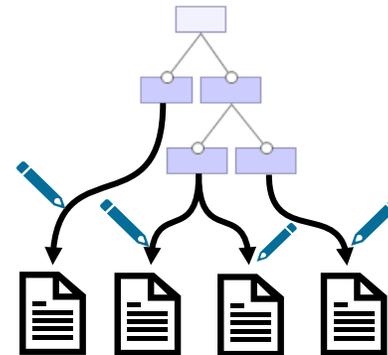
Consistency Checker



Ensure consistency between:

- Variability Models
- Delta Models
- Cross-Discipline Constraints

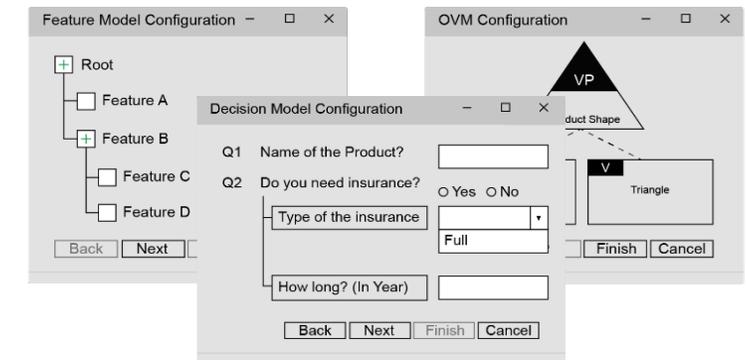
Artifacts Mapping Editor



Create a Graphical Editor to define mappings between:

- Delta Models
- Variation Points (e.g., Features)

Configuration Interfaces



Develop configuration interface for different variability models:

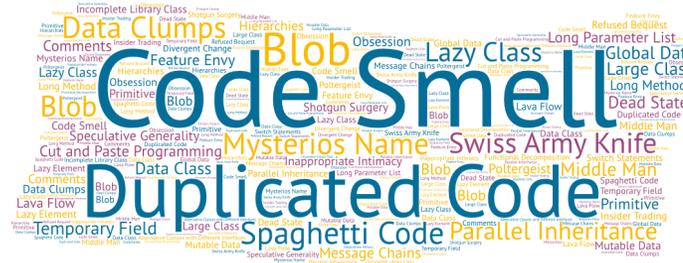
- Decision Models
- Orthogonal Variability Model (OVM)



Better (Control) Software Design

- Goals

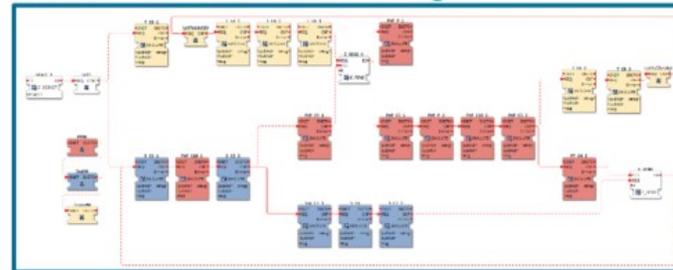
- Improve software quality
- Improve understandability
- Reduce maintenance effort



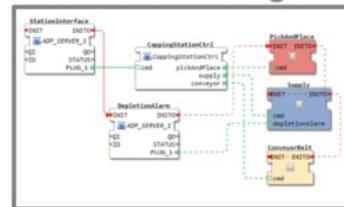
- Approach

- Bad smells
- Metrics
- Design patterns

Skill-based Distributed Design

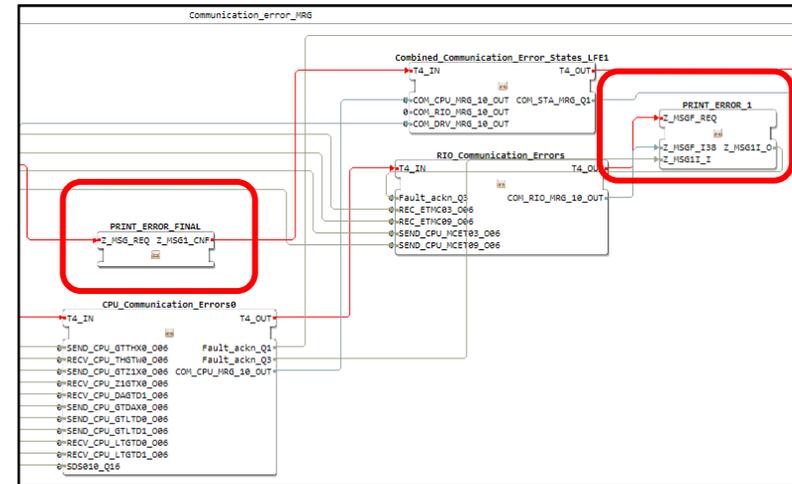


Hierarchical Design



IEC 61499 Bad Smells Catalog

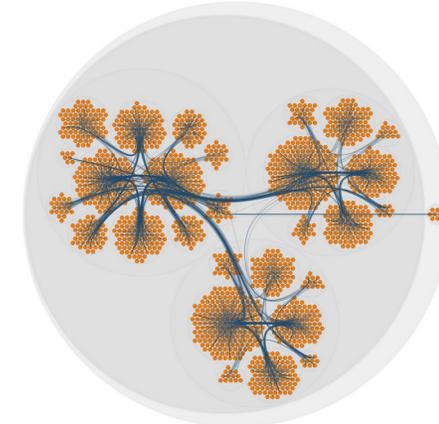
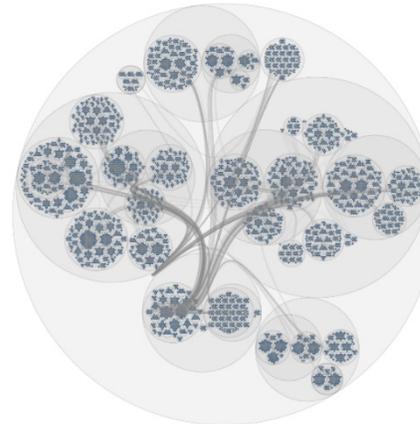
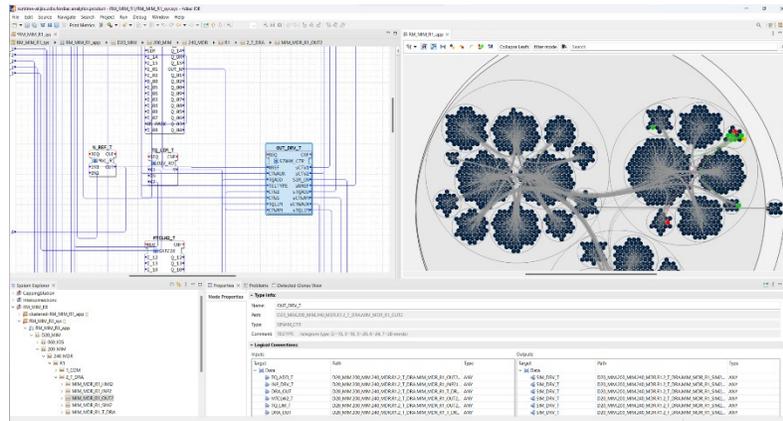
Name	Level	Description	Impl.
Duplicated Code	ALL	The same or similar code appearing more than once.	N
Long Algorithm	FB	An algorithm that is too long and complex.	N
Large Type	FB	An FB type that is too large and complex.	N
Large Interface	FB	Too many interface elements.	N
Divergent Change	FB	One change leading to many changes within the same FB type.	N
Shotgun Surgery	FB	One change leading to changes in many different FB types.	N
Feature Envy	ALL	An IEC 61499 component having high cohesion to another that should not be coupled tightly.	N
Data Clumps	FBN	A group of interface elements that always appear together.	N
Lazy Element	ALL	An IEC 61499 component without purpose (e.g., CFB only containing one FB).	N
Dead State	ECC	State (except start state) which does not have any input transitions or to which a path cannot be found from the EC initial state by following the directed links.	Y
Dead Transition	ECC	Transition with lower priority than the 1 transition condition	Y
Dead FB	FB	FB (except start FB) which does not have any input event connections.	Y
Terminal State	ECC	State that is reachable, but which does not have any outgoing EC transitions.	Y
Unused Event	FB	Event input/output of the FB type containing the ECC that is not used in any EC transitions.	Y
Unused Data	FB	When the particular input event is connected, the associated data input is unconnected or not configured.	Y
Mutable Data	ALG	The algorithm writes on a data input.	N
Dead Event	FB	An event that is not used in the transition condition of any stable ECC state and is thus always ignored.	N





Complexity/Modularization of Control Software

- **Goal: analyze/visualize modularization/complexity of (control) software**
- **Use Cases**
 - Understand structure of existing systems and component relations
 - Analyze modularization of existing systems (as input to improve/refactor)
 - Provide input for variability management/evolution (diffs of variants/versions...)





Software Engineering Tools for CPS

A USER
INTERFACE
IS LIKE A JOKE.
IF YOU HAVE TO
EXPLAIN IT, IT'S
NOT THAT
GOOD.

The screenshot shows the Eclipse IDE interface for a project named 'CappingStation'. The main window displays a state transition diagram for a conveyor system. The diagram includes states like 'INIT', 'running', 'stopped', and 'error', with transitions triggered by events like 'cmd.turnOn' and 'cmd.turnOff'. A 'ConveyorCtrl' block is highlighted in the diagram. The left sidebar shows a project tree with folders like 'CappingStationApp' and 'StationComponents'. The bottom panel shows a table of inputs and outputs for the selected component.

Inputs				Outputs			
Name	Type	Comment	Name	Type	Comment	Name	Type
1 cmd	AConveyor		1 motor	AMotor		2 inletBlocker	ABlocker
			3 mainBlocker	ABlocker		4 index	ACylinder

The screenshot shows the IDE interface for a motor control system. The top window displays a state transition diagram with states 'START', 'turnOn', 'running', 'turnOff', 'stopped', and 'error'. Transitions are triggered by events like 'cmd.turnOn', 'cmd.turnOff', and 'timer.Timeout'. The bottom window shows the algorithm code for the motor control.

```

ALGORITHM REQ
IF NOT init OR RST THEN
  init := BOOL#TRUE;
  in_last := IN;
  t_last := ULINT_TO_ULINT(TIME_IN_US_TO_ULINT(NOW_MONOTONIC()));
  i := REAL#0.0;
  tc := REAL#0.0;
ELSE
  (* read last cycle time in Microseconds *)
  tx := ULINT_TO_ULINT(TIME_IN_US_TO_ULINT(NOW_MONOTONIC()));
  tc := ULINT_TO_REAL(tx - t_last);
  t_last := tx;

  (* calculate proportional part *)
  p := KP * IN;

  (* run integrator *)
  i := (IN + in_last) * REAL#5.0E-7 * KI * tc + i;
  in_last := IN;

  (* calculate output Y *)
  Y := p + i;
  
```



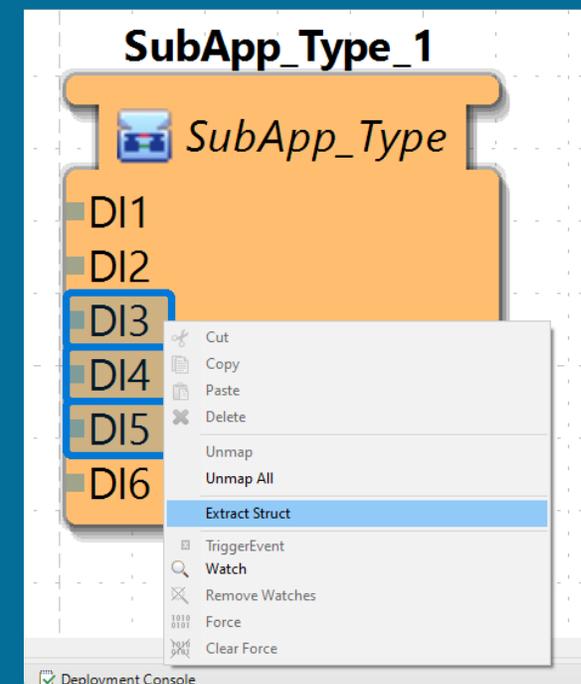
Implement a Smart Recommender System for Refactoring Operations in 4diac IDE

- Recommender systems are widely used in the entertainment sector. Like Netflix suggests the next movie to watch. So we want that the user saves time and the tool suggests features based on the user previous behavior
- **Goal:** Detect the users selection and provide appropriate suggestions for refactoring

Example:

4diac IDE offers the create struct feature which merges the pins and creates a datatype out of it.

So if the user manually merges the pins and creates the datatype manually the recommender should detect that and provide feedback to the user





Improving the Eclipse Platform

The screenshot displays the Eclipse IDE interface. The main editor shows the following Java code snippet:

```

@Override
public boolean keyPressed(final KeyEvent event) {
    final boolean modifierPressed = (event.stateMask & SWT.MODIFIER_MASK) != 0;
    currentKeyCode = event.keyCode;
    switch (event.keyCode) {
        case SWT.ARROW_DOWN:
            if (!modifierPressed) {
                getView().scrollByOffset(0, SCROLL_SPEED_Y);
                return true;
            }
            break;
        case SWT.ARROW_UP:
            if (!modifierPressed) {
                getView().scrollByOffset(0, -SCROLL_SPEED_Y);
                return true;
            }
            break;
        case SWT.ARROW_RIGHT:
            if (!modifierPressed) {
                getView().scrollByOffset(SCROLL_SPEED_X, 0);
                return true;
            }
            break;
        case SWT.ARROW_LEFT:
            if (!modifierPressed) {
                getView().scrollByOffset(-SCROLL_SPEED_X, 0);
                return true;
            }
            break;
        case SWT.PAGE_DOWN:
            if ((event.stateMask & SWT.SHIFT) != 0) {
                getView().scrollByOffset(getViewer().getFigureCanvasSize().x, 0);
            } else {
                getView().scrollByOffset(0, getViewer().getFigureCanvasSize().y);
            }
            return true;
        case SWT.PAGE_UP:
            if ((event.stateMask & SWT.SHIFT) != 0) {
                getView().scrollByOffset(-getViewer().getFigureCanvasSize().x, 0);
            } else {
                getView().scrollByOffset(0, -getViewer().getFigureCanvasSize().y);
            }
    }
}

```

The left sidebar shows the Project Explorer with the following structure:

- org.eclipse.fordiac.ide.gcf [org.eclipse.4diac]
 - JRE System Library [JavaSE-11]
 - Plug-in Dependencies
 - src
 - org.eclipse.fordiac.ide.gcf
 - org.eclipse.fordiac.ide.gcf.command
 - org.eclipse.fordiac.ide.gcf.dnd
 - org.eclipse.fordiac.ide.gcf.draw2d
 - org.eclipse.fordiac.ide.gcf.editors
 - org.eclipse.fordiac.ide.gcf.editparts
 - org.eclipse.fordiac.ide.gcf.figures
 - org.eclipse.fordiac.ide.gcf.filters
 - org.eclipse.fordiac.ide.gcf.handlers
 - AbstractZoomHandler.java
 - AdvancedGraphicalViewerKeyHan

The right sidebar shows the Outline view with the following structure:

- org.eclipse.fordiac.ide.gcf.handlers
 - AdvancedGraphicalViewerKeyHandler
 - SCROLL_SPEED_X: int
 - SCROLL_SPEED_Y: int
 - currentKeyCode: int
 - AdvancedGraphicalViewerKeyHandle
 - getCurrentKeyCode(): int
 - getView(): AdvancedScrollingGraph
 - keyPressed(KeyEvent): boolean
 - keyReleased(KeyEvent): boolean

The bottom status bar shows 17 errors, 2,485 warnings, and 948 others. The error list includes:

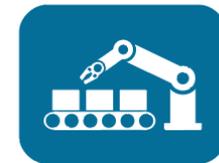
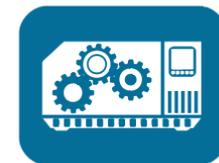
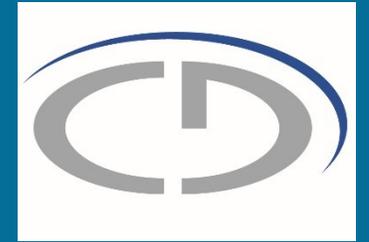
Description	Resource	Path
Couldn't resolve reference to EClassifier 'STVarGlobalDeclarationBlock'.	GlobalConstants.xtext	/org.eclipse.fo
Couldn't resolve reference to EClassifier 'STVarGlobalDeclarationBlock'.	GlobalConstants.xtext	/org.eclipse.fo

Teaching @ LIT CPS Lab



Christian Doppler Lab VaSiCS
LIT | Cyber-Physical Systems Lab
Johannes Kepler University Linz

JYU
LINZ INSTITUTE
OF TECHNOLOGY

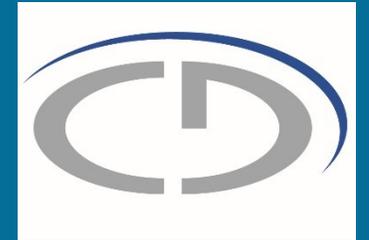


Teaching (Winter Semester)

- **Networked Embedded Systems** (VO/UE, Englisch) (Alois Zoitl, diverse)
 - Pflichtfach für ELIT, Mechatronik
- **Production Automation Systems** (VO, Englisch) (Alois Zoitl)
 - Wahlpflichtfach für Artificial Intelligence, Wahlfach für diverse Studienrichtungen
- **Practical Work in AI (Master)** (PR) (Alois Zoitl)
 - Pflichtfach für Master AI
- **Seminar in AI (Master)** (SE) (Alois Zoitl)
 - Pflichtfach für Master AI
- **Cloud Computing** (Rick Rabiser, Andreas Grimmer, Johannes Bräuer)
 - Wahlfach
- Project in **Computational Engineering** (PR) (Alois Zoitl, Rick Rabiser)
 - Wahlpflichtfach für Computer Science
- Project in **Software Engineering** (PR) (Alois Zoitl, Rick Rabiser)
 - Wahlpflichtfach für Computer Science
- **Projektpraktikum** (PR) (Bakk-Arbeit) (Rick Rabiser, Alois Zoitl)
 - Pflichtfach für Informatik
- **Master's Thesis Seminar** SS (SE) (Alois Zoitl, Rick Rabiser)
 - Begleitend zur Masterarbeit
- **Dissertantenseminar** Informatik (SE) (Alois Zoitl, Rick Rabiser)
 - Pflichtfach für Doktoratsstudium, Fach Informatik

Teaching (Summer Semester)

- **Algorithmen und Datenstrukturen (VO/UE)** (Rick Rabiser, div.)
 - Pflichtfach für ELIT, Mechatronik, Maschinenbau
- **Präsentations- und Arbeitstechnik (KV)** (Grünbacher, Kotsis, Rabiser, div.)
 - Pflichtfach für Informatik Rick Rabiser
- **Product Line Engineering (KV)** (Rick Rabiser, div.)
- **Production Automation Systems (UE)** (Alois Zoitl)
 - Wahlpflichtfach für Artificial Intelligence, Wahlfach für diverse Studienrichtungen
- **Networked Embedded Systems (PR)** (Alois Zoitl)
 - Pflichtfach für ELIT, Mechatronik
- **Parallel Computing (KV)** (Wolfgang Schreiner, Alois Zoitl)
 - Wahlpflichtfach für Computer Science
- Project in **Computational Engineering (PR)** (Alois Zoitl, Rick Rabiser)
 - Wahlpflichtfach für Computer Science
- Project in **Software Engineering (PR)** (Alois Zoitl, Rick Rabiser)
 - Wahlpflichtfach für Computer Science
- **Projektpraktikum (PR)** (Bakk-Arbeit) (Rick Rabiser, Alois Zoitl)
 - Pflichtfach für Informatik
- **Master's Thesis Seminar SS (SE)** (Alois Zoitl, Rick Rabiser)
 - Begleitend zur Masterarbeit
- **Dissertantenseminar Informatik (SE)** (Alois Zoitl, Rick Rabiser)
 - Pflichtfach für Doktoratsstudium, Fach Informatik



Thank you!

<https://www.jku.at/lit/cps-lab>



Prof. Rabiser / Prof. Zoitl
Christian Doppler Lab VaSiCS
LIT | Cyber-Physical Systems Lab
Johannes Kepler University Linz

