

Version Management and Conflict Detection across Tools in a (Software+) Engineering Environment

Thomas Moser Florian Waltersdorfer Dietmar Winkler Stefan Biffli

Christian Doppler Laboratory CDL-Flex

Institute of Software Technology and Interactive Systems (ISIS)

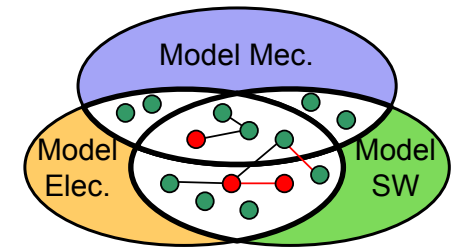
Vienna University of Technology

<http://cdl.ifs.tuwien.ac.at>



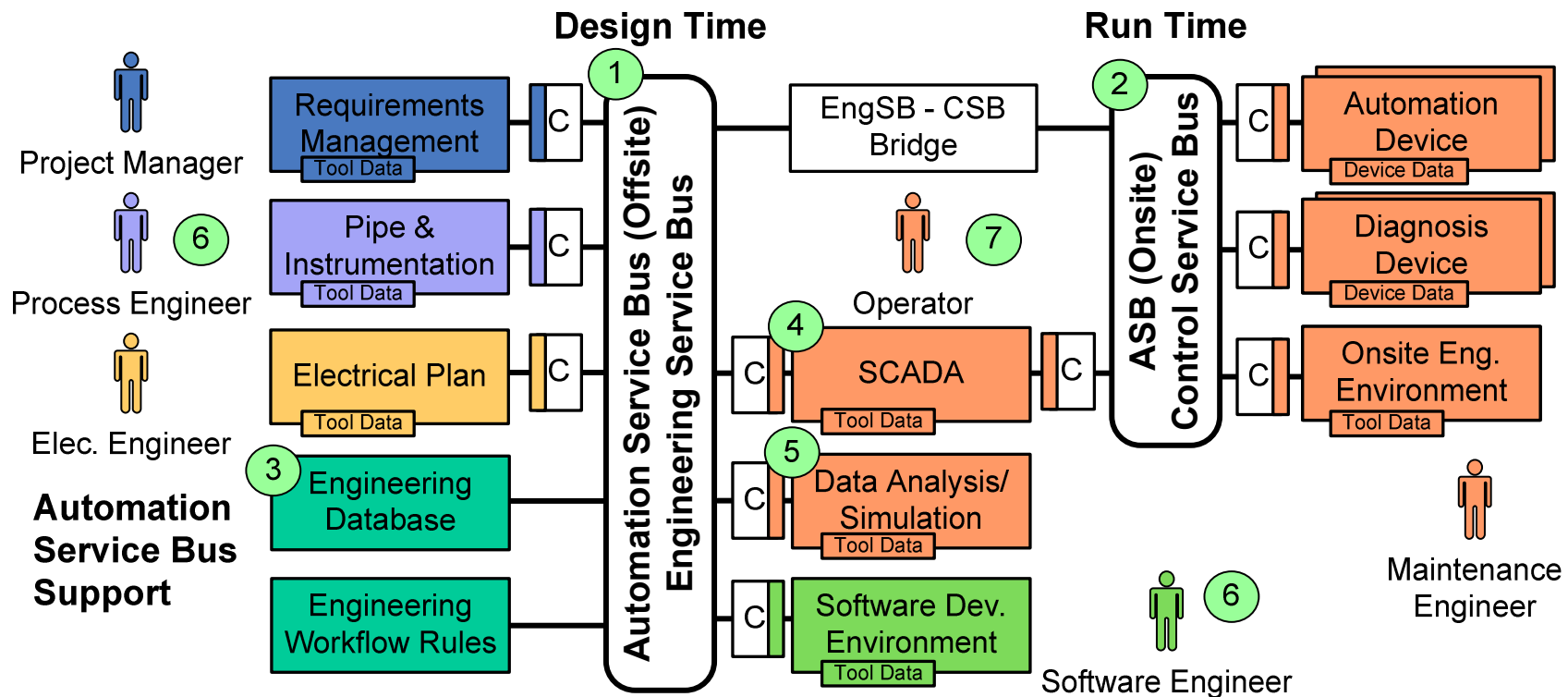
Context and Motivation

- Automation systems engineering projects
 - Contributions from several engineering disciplines.
 - Complex artifacts like mechanical, electrical, and software components and plans, which get updated concurrently.
- Version Management
 - Available for each individual engineering discipline.
 - Very little work on version management across semantically heterogeneous data models in engineering tools and projects.
- Quality Assurance (Change & Conflict Detection)
 - Selective QA activities in individual disciplines.
 - Challenge is to integrate quality assurance activities across disciplines and systems borders.



Automation Service Bus (ASB)

Goal: Approaches for the integration of software tools in automation engineering.



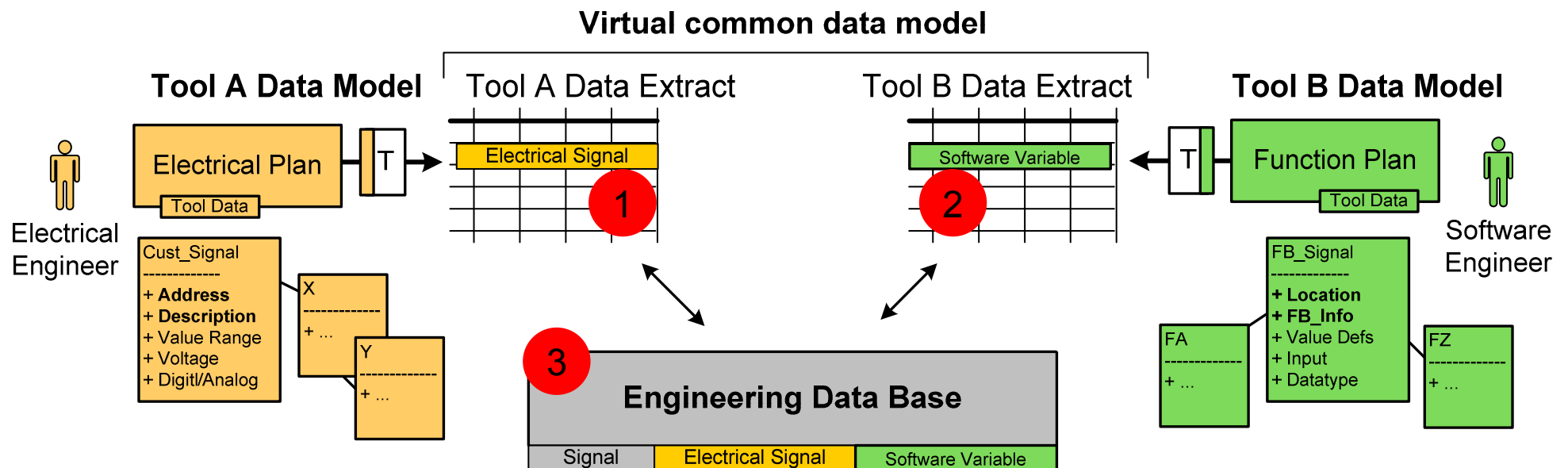
- Technical Integration: Engineering Service Bus (1), Control Service Bus (2).
- Semantic Integration: Engineering Database (3).
- Flexible integration of SCADA (4) with data analysis/simulation (5).
- Defect detection approaches for design time (6) and run time (7).

Foundation

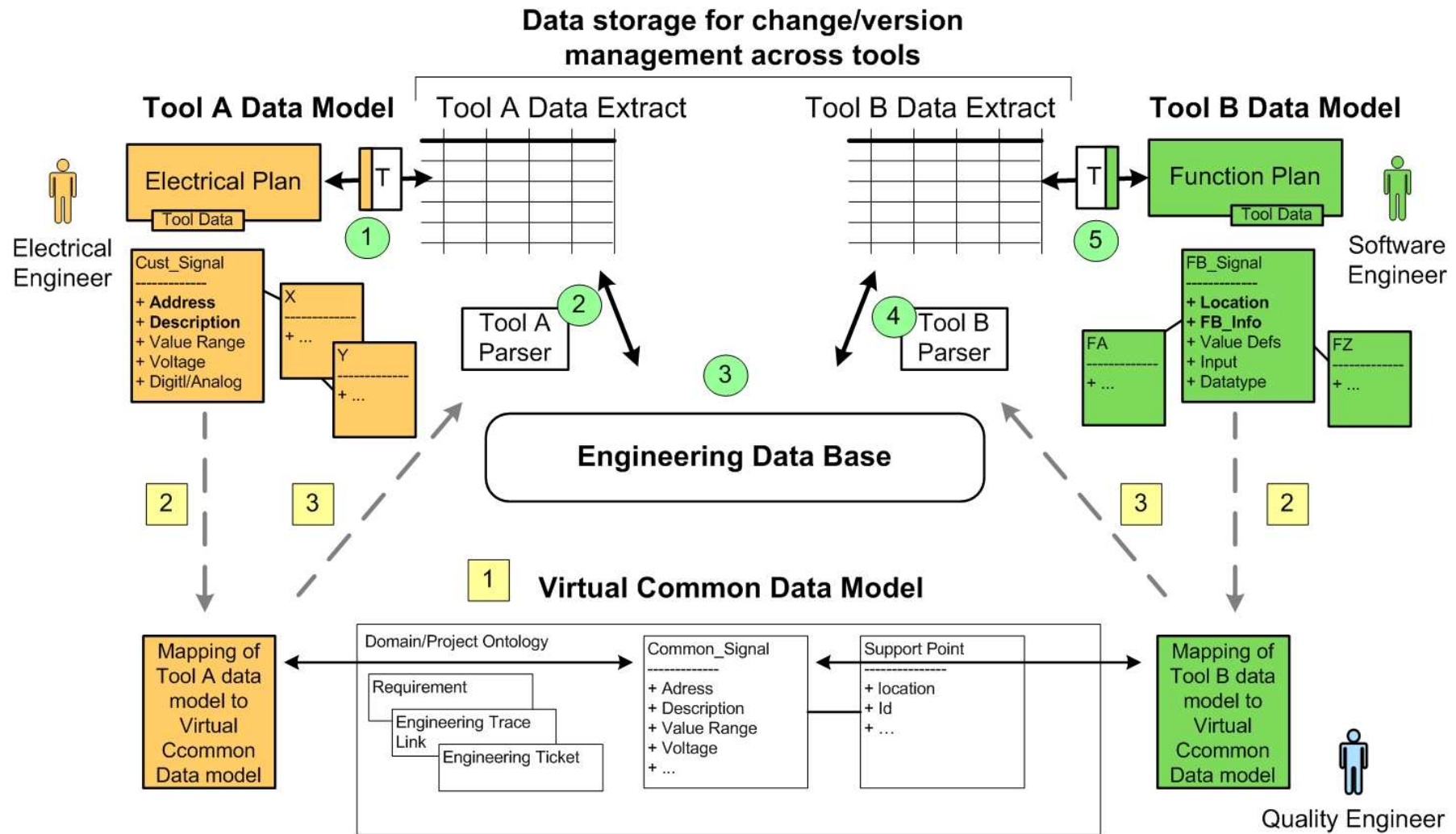
- The signal is a common concept for linking information between disciplines (e.g., mechanical interface, electrical signal (wiring), software I/O variable).


Challenges & Goals


- Consistent signal handling (e.g., up to 40,000 signals in power plants).
- Integration of signals from heterogeneous data models / tools (1) and (2).
- Version management of signal changes across engineering disciplines.
- Common concept based on semantic integration (3).



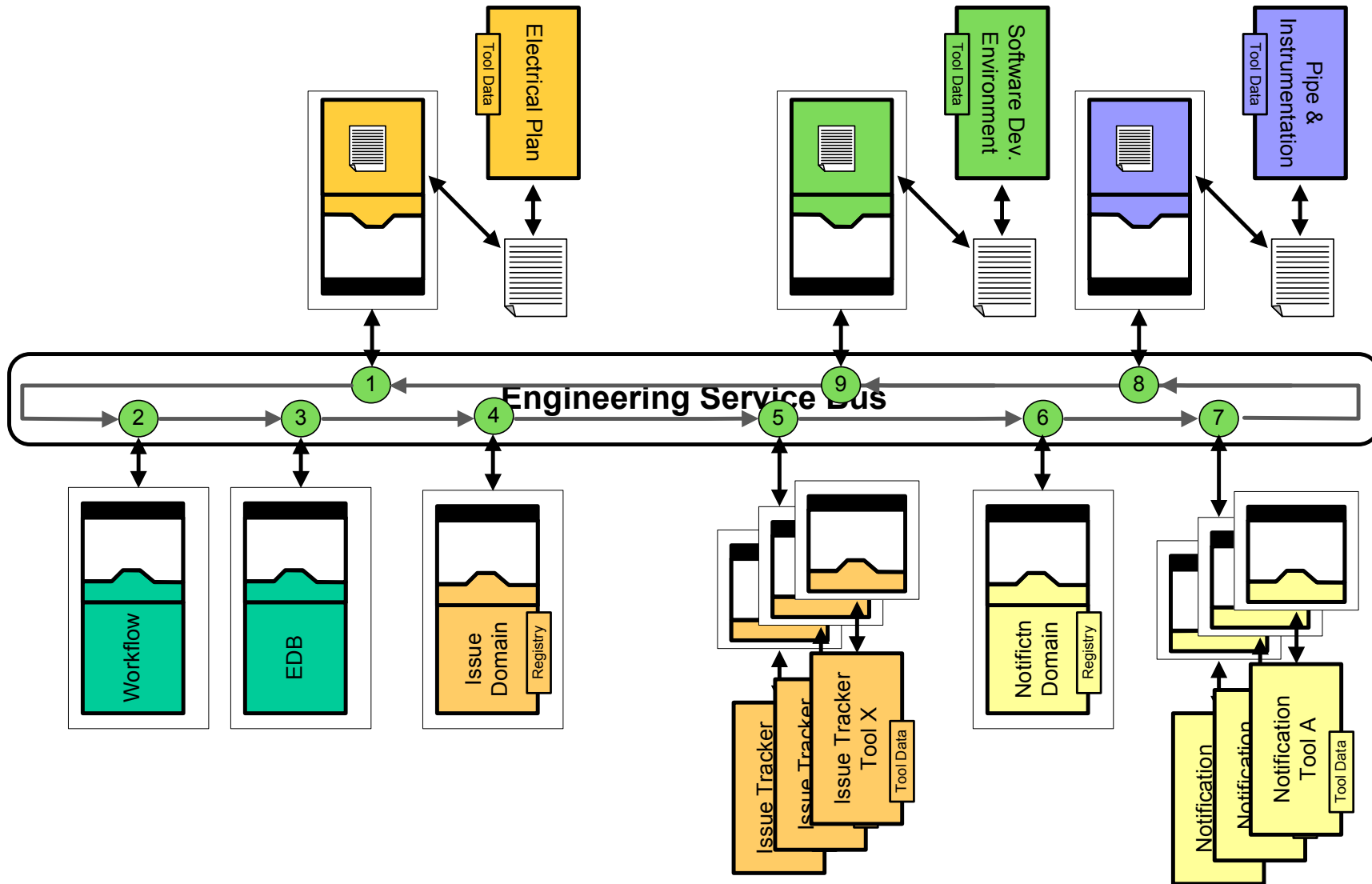
Virtual Common Data Model: Version Management across Tools



Numbered Circles: 
 Checkin,
 Checkout
 Version management

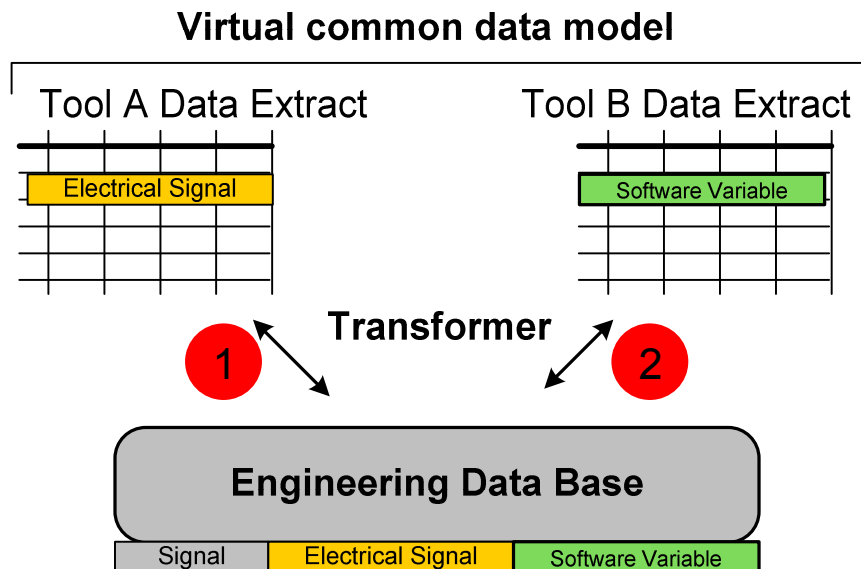
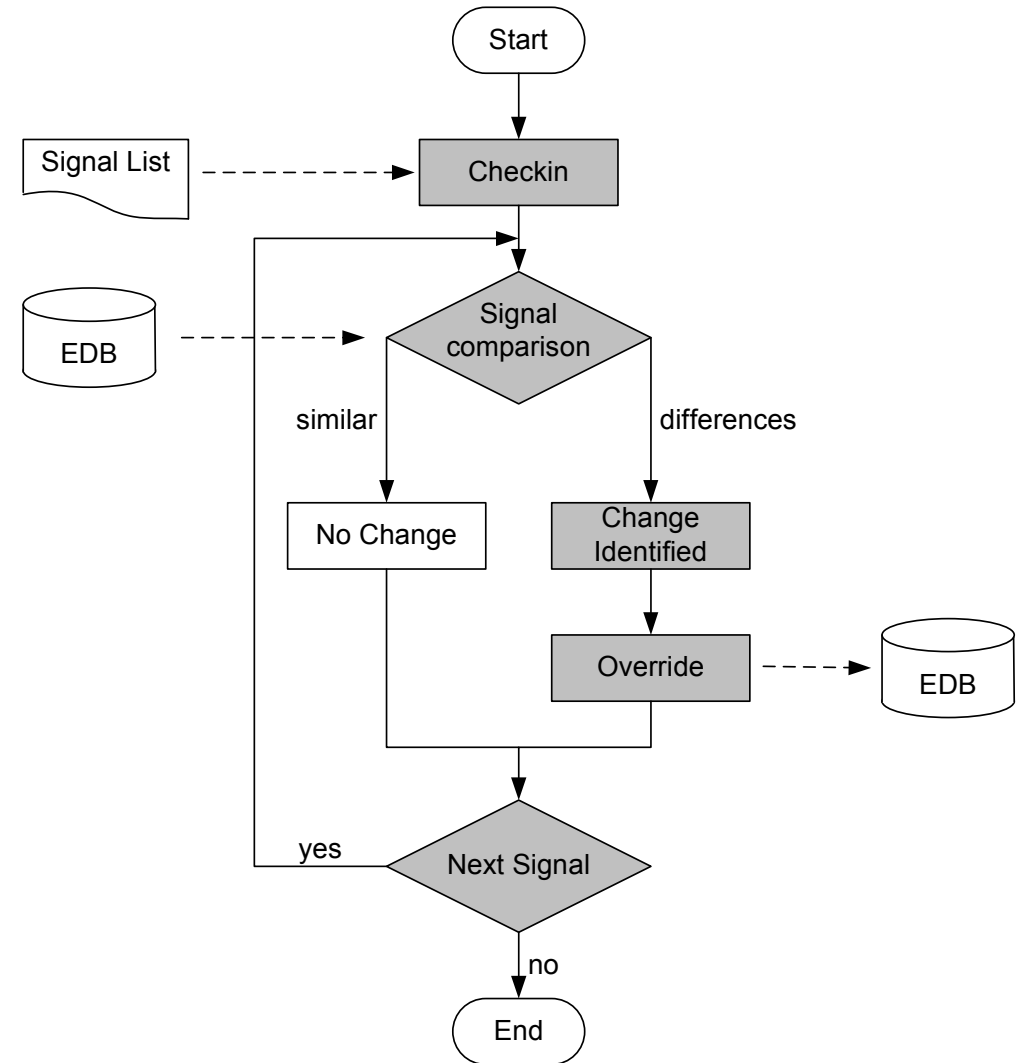
Numbered Squares: 
 Derive Virtual Common Data Model (VCDM)
 Derive Mapping from a tool to VCDM
 Configure parser with data mapping

Use Case – Signal Engineering of Hydro Power Plant Systems Integrators



Basic Signal Check-In Workflow

- Check-In of new signal list.
- Signal comparison with EDB.
- Pass new signals / unchanged signals to EDB.
- Manual confirmation of changed signals and override signal in EDB.



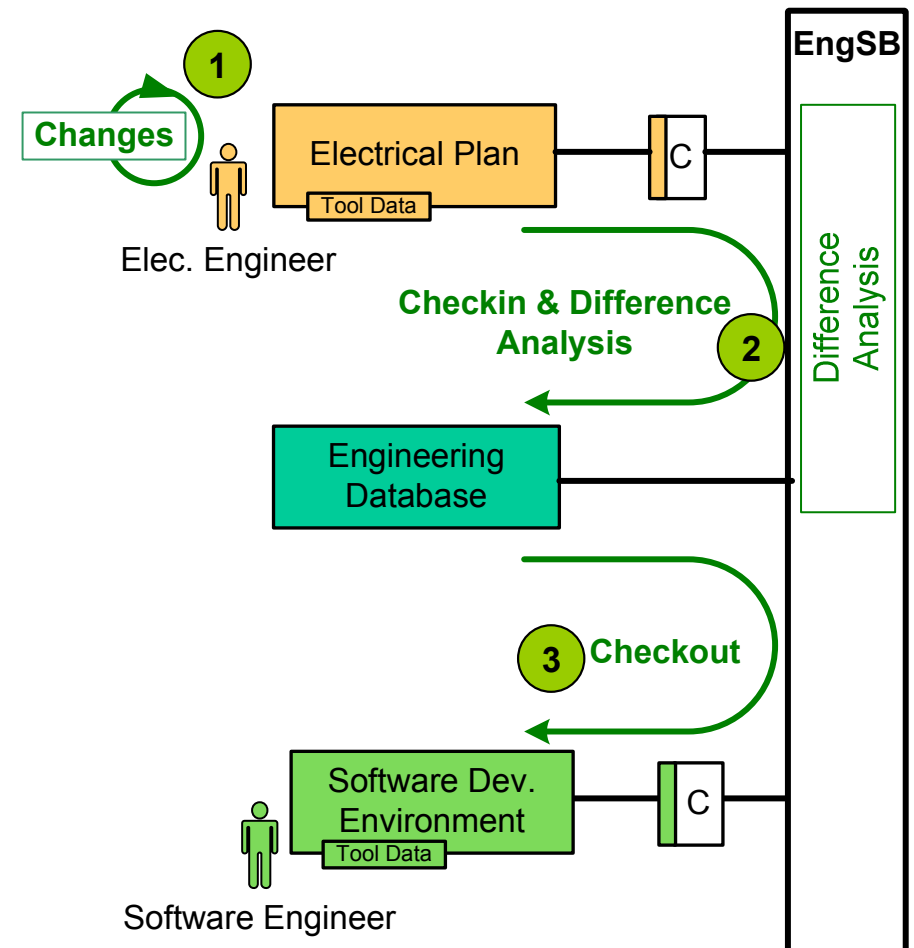
Signal Changes Across Tools and Disciplines

■ Challenges and Goals

- Merge changes between signals coming from different disciplines.

■ Conceptual Approach

1. Execute Changes.
2. Check-In and merge changes with Engineering Database
 - Conflicts can be changes semi-automatically.
 - Engineering tickets and notification in case of critical changes and conflicts (e.g., removed signals).
3. Check-Out merged signal lists.



- Check in: Status of Imported Signals

Merge Signals (Signals are replaced on default)

view new signals (0) | view unchanged signals (30) | **view conflicts (12)**

Tick checkboxes to the left of the new value in order to update a property.

Show only conflicts.

Different Views

- New Signals
- Unchanged Signals
- Changes / Conflicts

- Highlight & Resolve Differences

| Signal | Update whole row | | channelName | inputOutputModule | functionTextOne | region | projectId | customer |
|--------|--------------------------|--------------------------|-------------|-------------------|---|--------|-----------|----------|
| 0 | keep all | old value: new value: | 6 | 3 | H_U#p 110VDC High oil pres pump feeder (Q12 F71 F72) ready <input checked="" type="checkbox"/> H_U#p 110000VDC High oil pres pump feeder (Q12 F71 F72) ready | Tyrkie | Kandil | Sabanci |
| 0 | keep all | old value: new value: | 10 | 7 | H_U#p St Service MV cubicle HV fuse fault | Tyrkie | Kandil | Sabanci |

Conflicts

- Old & New values
- Selection and Notification
- Confirmation / Change & Conflict Resolution

Pilot Application: Signal Browser

- View signals in system topology.
- Identify and view of signal information.
- Export of result set and rework in spreadsheets.
- Candidate use case for next steps: Navigation to source data of signals in various tools (cross references).

Show/hide individual data fields

Queries for subsystem selection



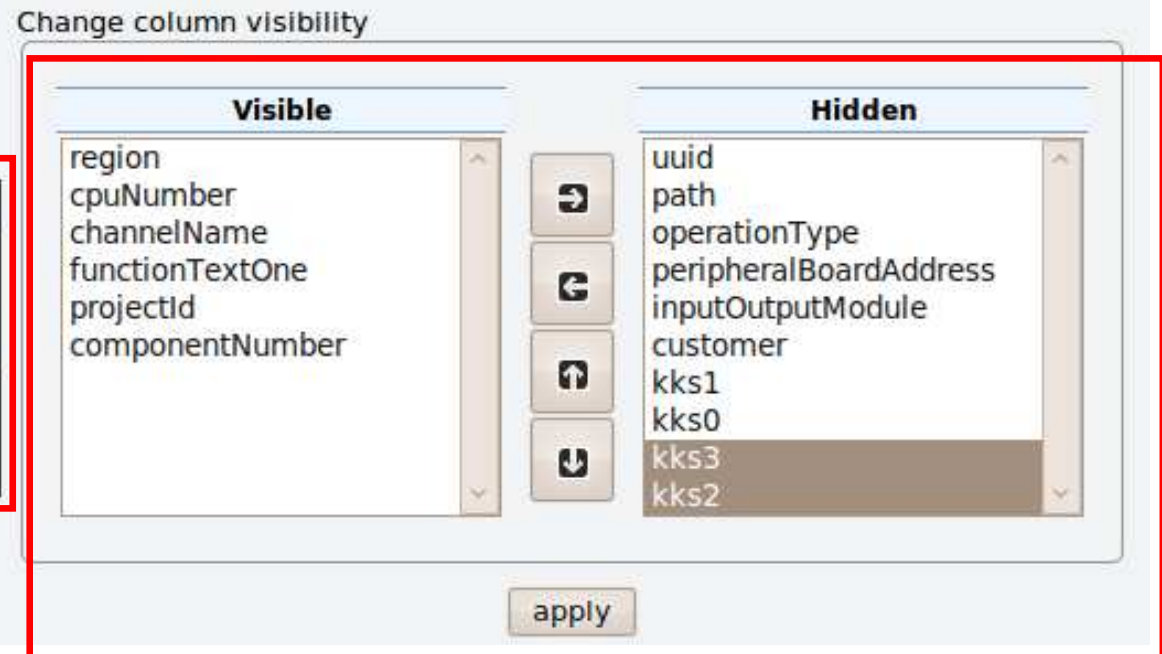
Query

Pathselection:





Keyselection: [submit](#)

project1

- B
- 040



Change column visibility

| Visible | | Hidden |
|-----------------|---|------------------------|
| region |     | uuid |
| cpuNumber | | path |
| channelName | | operationType |
| functionTextOne | | peripheralBoardAddress |
| projectId | | inputOutputModule |
| componentNumber | | customer |
| | kks1 | |
| | kks0 | |
| | kks3 | |
| | kks2 | |

[apply](#)

Pilot Application: Signal Browser

- View signals in system topology.
- Identify and view of signal information.
- Export of result set and rework in spreadsheets.
- Candidate use case for next steps: Navigation to source data of signals in various tools (cross references).

Result Set

export signals

Showing 1 to 50 of 347 << < 1 2 3 4 5 6 7 > >>

| line | region | cpuNumber | channelName | functionTextOne | projectId | componentNumber |
|------|--------|-----------|-------------|--|-----------|-----------------|
| 1 | | 02 | 10 | Drainage pit 1 - drainage pump 2 temperature | project1 | 040 |
| 2 | | 02 | 11 | Drainage pit 1 - jet pump control valve - open | project1 | 040 |
| 3 | | 02 | 10 | Drainage pit 2 - drainage pump 2 temperature | project1 | 040 |
| 4 | | 02 | 11 | Drainage pit 2 - jet pump control valve - open | project1 | 040 |
| 5 | | 02 | 12 | Drainage pit 1 - jet pump control valve - closed | project1 | 040 |

Pilot Application: History of Signal Data Check-Ins

Five most modified signals

customer/project1/turbine/Auxillary_Rack/CPU_2/Channel_1/Pin_3
customer/project1/control_board/Main_Rack/CPU_1/Channel_2/Pin_1
customer/project1/turbine/Auxillary_Rack/CPU_2/Channel_1/Pin_3
customer/project1/cooler/Main_Rack/CPU_3/Channel_1/Pin_3
customer/project1/turbine/Auxillary_Rack/CPU_1/Channel_6/Pin_2

Basic statistics on most frequently changed signals

Revision 127

Commit via Hydro-EDB API
draft for new turbine fallback wiring <admin@ahy.com>
Thu., 9. Dec. '10 - 2:01
25 added, 2 modified, 0 deleted
[more...](#)

Revision 126

Commit via Hydro-EDB API
Stress Test feedback <admin@ahy.com>
Thu., 9. Dec. '10 - 2:01
3 added, 2 modified, 5 deleted
[more...](#)

Check in history

Detailed Check-In Information

Committer: admin <admin@ahy.com>
Author: admin <admin@ahy.com>
Time: Tue., 19. Oct. '10 - 3:16
Message:
Commit via Hydro-EDB API
Previous Checkins

- 0f23ff9df702b48a8dfdd2a31a7b2407ab498c15

Summary: 2 added, 15 modified, 0 deleted

Added

- customer/project1/B/010/01/01/1/02/6f730a33-a49f-4a3e-a7c3-8f908831ee1a
- customer/project1/B/010/02/01/1/02/fb8a688a-ac30-4c2f-8eb3-07cfd1ced621

Modified

- customer/project1/B/010/01/01/2/01/eb9a636c-6bd6-47e4-bdbf-f4e3e693d3c4
- customer/project1/B/010/01/01/3/00/9723b660-0a9b-4914-a715-1389663fc0cd
- customer/project1/B/010/01/01/3/01/147a6618-2eef-4b13-9d63-b7e0337790fc

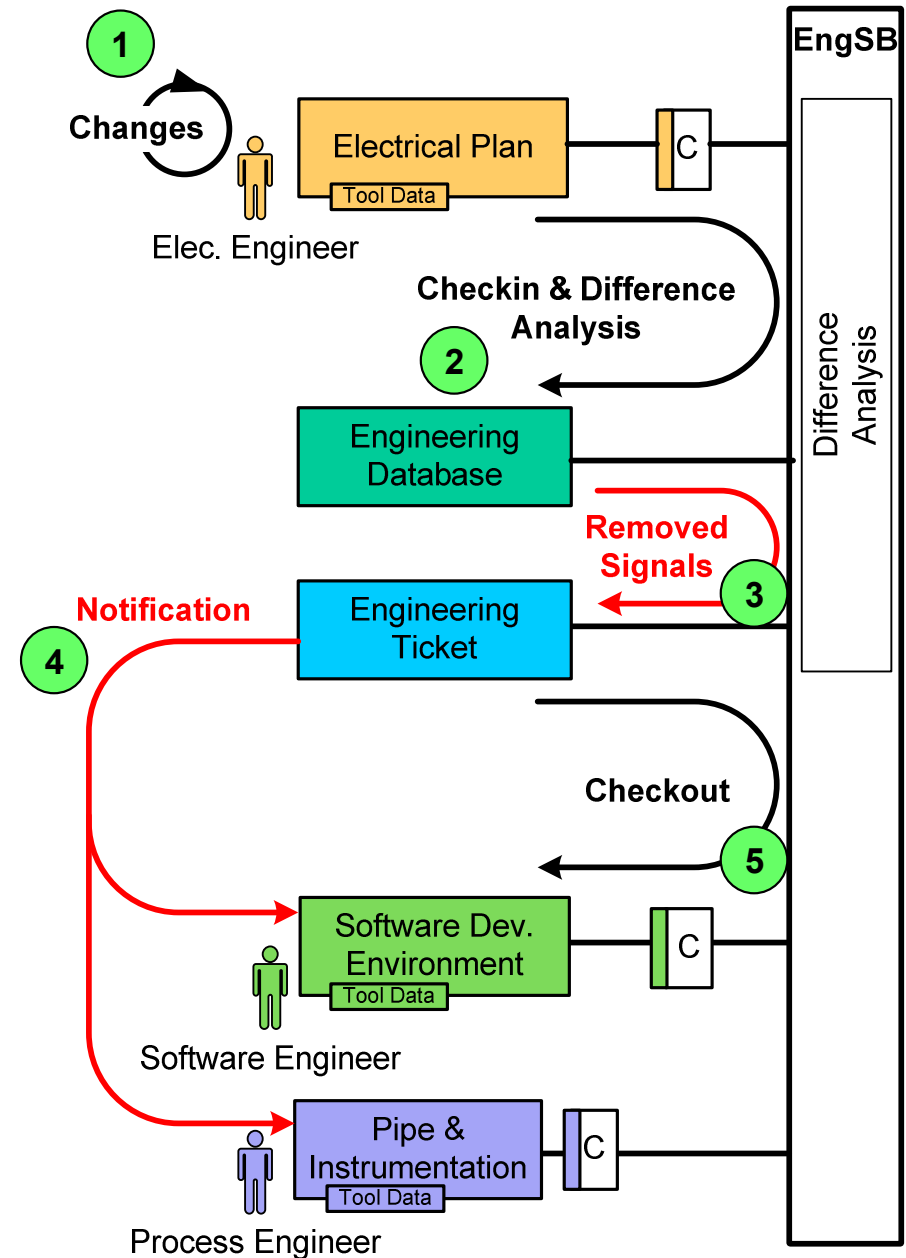
UC: Signal Deletion with Engineering Tickets

Challenges and Goals

- Some conflicts cannot be resolved during check-in, e.g., removed signals
- Notification required to minimize surprises in the engineering team

Conceptual Approach

1. Execute Changes
2. Conduct Difference Analysis
3. Identify “Removed Signals”
→ generate Engineering Ticket
4. Notify (multiple) related stakeholders
5. Checkout



Prototype: Engineering Ticket Overview



- **Challenges and Goals:**

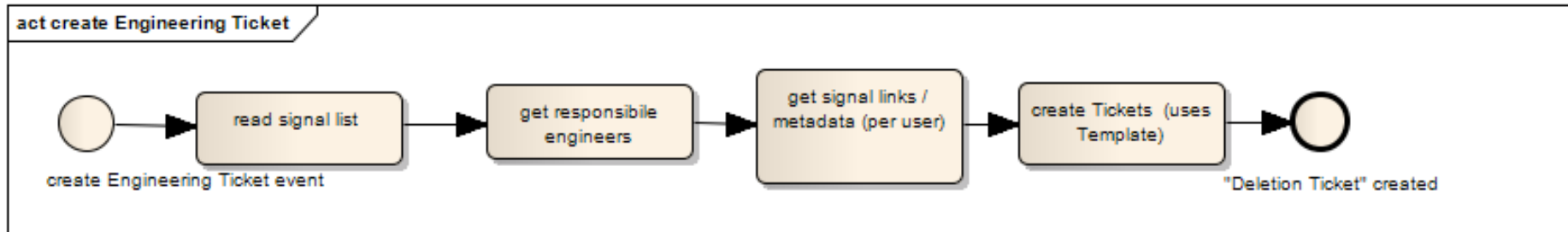
- Notification of stakeholders (e.g., warning on deleted signals)
- Ensure the correct process steps to deal with “deleted signals”:
Clear status of process

- **Approach**

- Engineering Ticket: “Change Request” that holds all relevant information for the roles involved.
- Allows tracking the process status
- Minimizes searching in documents

| Ticket | Summary | Component | Status | Resolution | Type | Priority | Owner | Modified |
|--------|--|----------------|----------|--------------------|-------------------------|----------|-----------------------|----------|
| #1 | Signal 2345-FDCB-1241 removed | Generator | new | | review (signal deleted) | major | florian.waltersdorfer | 04/08/10 |
| #3 | Signal 9537-A4DJ-2341 removed | Turbine2 | assigned | | review (signal deleted) | major | stefan.biffli | 04/08/10 |
| #8 | Signal 4232-FNXX-3283 changed | Turbine1 | accepted | | approve (signal change) | major | peter.fruehwirt | 04/08/10 |
| #9 | Signal 1232-UFEW-9231 changed | Generator | new | | review (signal deleted) | major | stefan.biffli | 04/08/10 |
| #12 | Signals changed (4 unapproved) | Schaltzentrale | new | | approve (signal change) | major | dietmar.winkler | 17/09/10 |
| #11 | Signal deletion by florian.waltersdorfer (1 signals) | Turbine | closed | clear for deletion | review (signal deleted) | major | dominik.hofer | 24/09/10 |
| #10 | Signal deletion by florian.waltersdorfer (2 signals) | Turbine | closed | request for change | review (signal deleted) | major | dominik.hofer | 24/09/10 |
| #7 | Signal 9324-FWDF-2312 changed | Generator | closed | rejected | review (signal deleted) | major | peter.fruehwirt | 04/08/10 |
| #6 | Signal 2333-WETD-9452 changed | Schaltzentrale | closed | approved | approve (signal change) | major | peter.fruehwirt | 04/08/10 |
| #5 | Signal 9122-UWDZ-2332 removed | Schaltzentrale | closed | clear for deletion | review (signal deleted) | major | florian.waltersdorfer | 04/08/10 |
| #4 | Signal 2312-ZWDA-1237 removed | Schleuse | closed | rejected | review (signal deleted) | major | stefan.biffli | 04/08/10 |
| #2 | Signal 2781-ADEI-1325 changed | Generator | closed | rejected | approve (signal change) | major | peter.fruehwirt | 04/08/10 |

Prototype: "Deletion" Engineering Ticket logi.cals®



■ Pre-Defined Ticket Information

Ticket #10 (closed review (signal deleted): request for change)

Modify ↓

Signal deletion by florian.waltersdorfer (2 signals) Opened 3 months ago
Last modified 0 seconds ago

| | | | |
|--------------|---------------------------------------|------------|-----------------------------------|
| Reported by: | florian.waltersdorfer | Owned by: | dominik.hofer |
| Priority: | major | Component: | Turbine |
| Keywords: | | Cc: | michael.petritsch |

Description (last modified by hydro) (diff)

Signal Main_Rack/CPU_1/Channel_4/Pin_2 (id: 2d9e6e...) has been deleted. Reply
Type: analog
Text: U1 - TEMP. STATORWINDING/phase U/ centre
KKS: G-MKA20-CT001-B01

Further Information

- ↳ <http://www.andritz.com/de/hydro/boyabat/links/eplan/2d9e6eb2-7eb6-4dac-b75b-b022b77618d7>

Signal Auxillary_Rack/CPU_2/Channel_1/Pin_3 (389459...) has been deleted.
Type: digital
Text: 400 VAC Main distribution, busbar A, undervoltage
KKS: G-MKA20-CL001-S01

Further Information

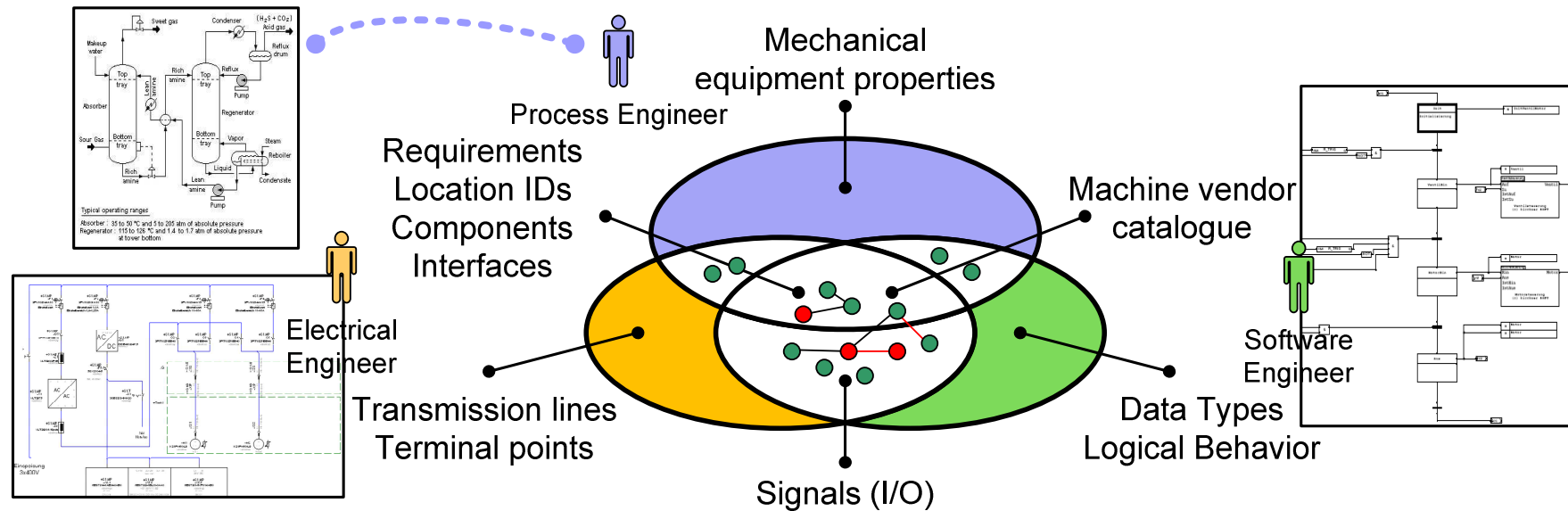
- ↳ <http://www.andritz.com/de/hydro/boyabat/links/eplan/38945975-a91b-46d6-81de-d3a2119d2967>

Data Source:
Project Role Concept

Data source:
Engineering Database

Quality Assurance & Defect Detection in Engineering Models across Tools

Use of common concepts in models across engineering disciplines



Defect type examples

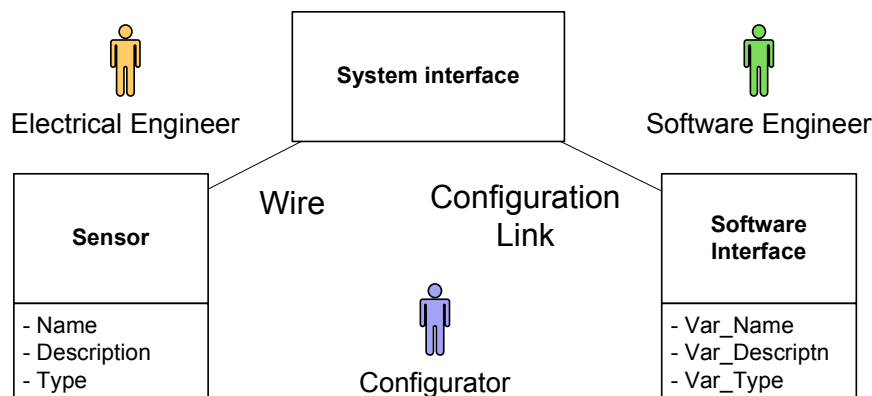
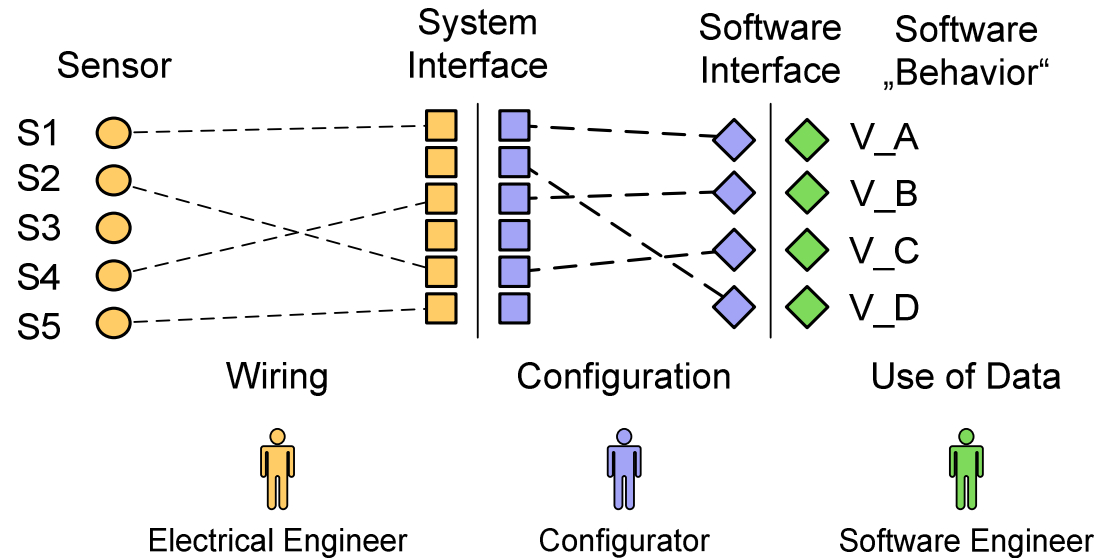
- Missing, wrong, inconsistent model elements or relationships
- **Conflicts from changes** of overlapping model elements
- Run-time violation of model constraints

Defect detection approaches

- Review of overlapping model parts
- Automated check of model assertions (syntactic and semantic)
- **Change conflict detection** and resolution
- Derivation of run-time assertions

End-to-End Test Across Engineering Models

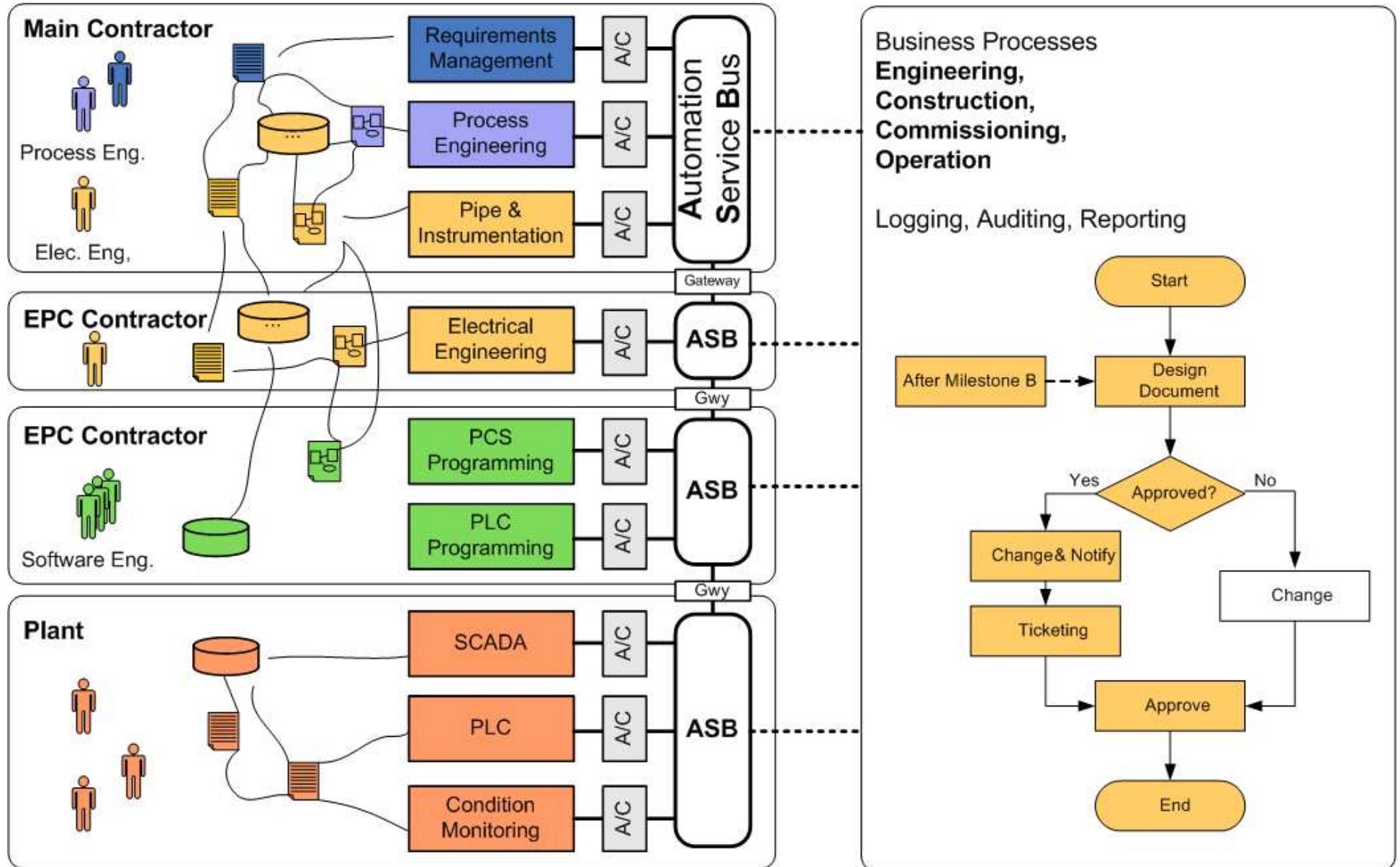
Use of common concepts in models across engineering disciplines



End-to-End Analysis

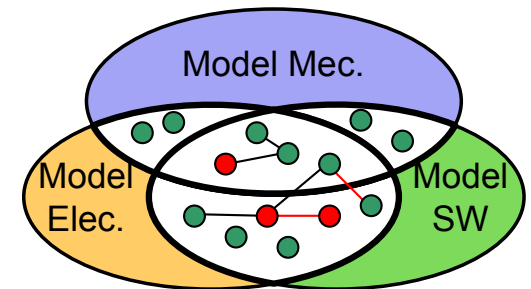
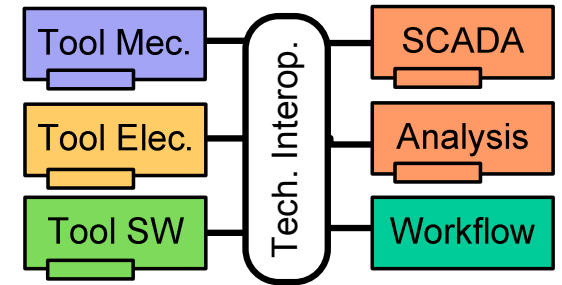
- List of sensor name/description/type with Variable name/description/type
- Warnings for incomplete chains between variables and sensors

Engineering Process Automation Concept



Conclusion and Further Work

- Automation systems engineering projects
 - Contributions from several engineering disciplines
 - Need for version management across semantically heterogeneous data models in engineering tools and projects
- Automation Service Bus (ASB) and Engineering Database (EDB) concept enables
 - Version management
 - Change & conflict detection and resolution
 - Integrated quality assurance activities
- Further research work
 - Identify new use cases from heterogeneous application domains.
 - Identify candidate industry partners for research prototype development.



Thank you ...



Version Management and Conflict Detection across Tools in a (Software+) Engineering Environment

Thomas Moser¹, Florian Waltersdorfer¹, Dietmar Winkler^{1,2}, Stefan Biffli¹

¹Christian Doppler Laboratory CDL-Flex
Institute of Software Technology and Interactive Systems, TU Vienna

²logi.cals automation solutions & services GmbH

<http://cdl.ifs.tuwien.ac.at>
Dietmar.Winkler@tuwien.ac.at