Interdisciplinary Systems Development Projects: Change Management across Disciplines and Tools

Dietmar Winkler  Stefan Biffl
Christian Doppler Laboratory “Software Engineering Integration for Flexible Automation Systems” (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.
  - Distributed development projects.
  - Complex artifacts like mechanical, electrical, and software components and plans, which get updated concurrently.

- Change and Version Management
  - Available for each individual engineering discipline.
  - Very little work on change and management across semantically heterogeneous data models in engineering tools and projects.

- Challenge
  - Efficient change management activities embedded within an engineering process across disciplines, data models, and tools.
Concurrently changes in distributed environments require efficient change management approaches (1).

Efficient synchronization mechanisms (2) enable cross-disciplinary change management based the Engineering Service Bus Platform.
Agenda

- **Efficient change management process** across disciplines, data models, and tools:
  - Identification of common concepts in individual disciplines to link domain-specific data models.
  - Establishing a virtual common data model for efficient and effective data exchange approaches.
  - Establishing a change management process approach across disciplines and tool borders.

- **Prototype implementation**
  - Feasibility study of the integrated change management approach at a hydro power plant systems integrator.
  - Measurement of processes for verification and validation.

- **Added value component**
  - Engineering cockpit for project monitoring and control.
Common Concepts: Signals & Signal Engineering

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

Tool A Data Model
- Electrical Plan
  - Tool Data
  - Electrical Signal
  - Software Variable
  - Cust_Signal
    - + Address
    - + Description
    - + Value Range
    - + Voltage
    - + Digital/Analog

Tool A Data Extract
- Tool B Data Extract
  - Tool B Data Model
    - Function Plan
    - Tool Data
    - FB_Signal
      - + Location
      - + FB_Info
      - + Value Defs
      - + Input
      - + Datatype

Engineering Data Base
- Signal
- Electrical Signal
- Software Variable
Virtual Common Data Model: Change & Version Management across Tools

Data storage for change/version management across tools

Tool A Data Model

Tool A Data Extract

Tool B Data Extract

Tool B Data Model

Electrical Plan

Cust_Signal
+ Address
+ Description
+ Value Range
+ Voltage
+ Digital/Analog

Tool A Parser

Tool B Parser

Function Plan

FB_Signal
+ Location
+ FB_Info
+ Value Defs
+ Input
+ Datatype

FA
+ ...

FZ
+ ...

Engineering Data Base

Virtual Common Data Model

Mapping of Tool A data model to Virtual Common Data model

Domain/Project Ontology

Requirement
Engineering Trace
Link
Engineering Ticket

Common_Signal
+ Address
+ Description
+ Value Range
+ Voltage
+ ...

Support Point
+ location
+ Id
+ ...

Mapping of Tool B data model to Virtual Common Data model

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
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.
Pilot Application: Change & Conflict Identification & Resolution

- **Check in: Status of Imported Signals**

![Merge Signals (Signals are replaced on default)](image)

- **Different Views**
  - New Signals
  - Unchanged Signals
  - Changes / Conflicts

- **Highlight & Resolve Differences**

![Highlight & Resolve Differences](image)

**Conflicts**
- Old & New values
- Selection and Notification
- Confirmation / Change & Conflict Resolution
Pilot Application: 
History of Signal Data Check-Ins

- Basic statistics on most frequently changed signals
- Detailed information on
  - Previous checkins
  - Summary of current checkin (e.g., added signals, removed signals, and modified signals)
Notification based on Changes: 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 Implementation:
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

<table>
<thead>
<tr>
<th>Ticket</th>
<th>Summary</th>
<th>Component</th>
<th>Status</th>
<th>Resolution</th>
<th>Type</th>
<th>Priority</th>
<th>Owner</th>
<th>Modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>Signal 2345-FDCB-1241 removed</td>
<td>Generator</td>
<td>new</td>
<td>review (signal deleted)</td>
<td>major</td>
<td>florian.waltersdorfer</td>
<td>04/08/10</td>
<td></td>
</tr>
<tr>
<td>#3</td>
<td>Signal 9537-A4D1-2341 removed</td>
<td>Turbine2</td>
<td>assigned</td>
<td>review (signal deleted)</td>
<td>major</td>
<td>stefan.biffl</td>
<td>04/08/10</td>
<td></td>
</tr>
<tr>
<td>#8</td>
<td>Signal 4232-FNXX-3283 changed</td>
<td>Turbine1</td>
<td>accepted</td>
<td>approve (signal change)</td>
<td>major</td>
<td>peter.fruehwirt</td>
<td>04/08/10</td>
<td></td>
</tr>
<tr>
<td>#9</td>
<td>Signal 1232-UFEW-9231 changed</td>
<td>Generator</td>
<td>new</td>
<td>review (signal deleted)</td>
<td>major</td>
<td>stefan.biffl</td>
<td>04/08/10</td>
<td></td>
</tr>
<tr>
<td>#12</td>
<td>Signals changed (4 unapproved)</td>
<td>Schaltzentrale</td>
<td>new</td>
<td>approve (signal change)</td>
<td>major</td>
<td>dietmar.winkler</td>
<td>17/09/10</td>
<td></td>
</tr>
<tr>
<td>#11</td>
<td>Signal deletion by florian.waltersdorfer (1 signals)</td>
<td>Turbine</td>
<td>closed</td>
<td>clear for deletion</td>
<td>review (signal deleted)</td>
<td>major</td>
<td>dominik.hofer</td>
<td>24/09/10</td>
</tr>
<tr>
<td>#10</td>
<td>Signal deletion by florian.waltersdorfer (2 signals)</td>
<td>Turbine</td>
<td>closed</td>
<td>request for change</td>
<td>review (signal deleted)</td>
<td>major</td>
<td>dominik.hofer</td>
<td>24/09/10</td>
</tr>
<tr>
<td>#7</td>
<td>Signal 9324-FWDF-2312 changed</td>
<td>Generator</td>
<td>closed</td>
<td>rejected</td>
<td>review (signal deleted)</td>
<td>major</td>
<td>peter.fruehwirt</td>
<td>04/08/10</td>
</tr>
<tr>
<td>#6</td>
<td>Signal 2333-WETD-9452 changed</td>
<td>Schaltzentrale</td>
<td>closed</td>
<td>approved</td>
<td>approve (signal change)</td>
<td>major</td>
<td>peter.fruehwirt</td>
<td>04/08/10</td>
</tr>
<tr>
<td>#5</td>
<td>Signal 9122-UWDZ-2332 removed</td>
<td>Schaltzentrale</td>
<td>closed</td>
<td>clear for deletion</td>
<td>review (signal deleted)</td>
<td>major</td>
<td>florian.waltersdorfer</td>
<td>04/08/10</td>
</tr>
<tr>
<td>#4</td>
<td>Signal 2312-ZWDA-1237 removed</td>
<td>Schleuse</td>
<td>closed</td>
<td>rejected</td>
<td>review (signal deleted)</td>
<td>major</td>
<td>stefan.biffl</td>
<td>04/08/10</td>
</tr>
<tr>
<td>#2</td>
<td>Signal 2781-ADEI-1325 changed</td>
<td>Generator</td>
<td>closed</td>
<td>rejected</td>
<td>approve (signal change)</td>
<td>major</td>
<td>peter.fruehwirt</td>
<td>04/08/10</td>
</tr>
</tbody>
</table>
Prototype Implementation: 
“Deletion” Engineering Ticket

- Pre-Defined Ticket Information

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

**Signal deletion by florian.waltersdorfer (2 signals)**

- Reported by: florian.waltersdorfer
- Owned by: dominik.hofor
- Priority: major
- Component: Turbine
- Keywords: 
- Cc: michael.patrichts

**Description (last modified by hydro):**

- Signal Main_Rack/CPU_1/Channel_4/Pin_2 (id: 2d9e6e... has been deleted.
- Type: analog
- Text: U1 - TEMP.STATOR.WINDING/phase U/centre
- KKS: MAKA20-CT001-B01

**Further Information**

- [http://www.andritz.com/de/hydro/boyabat/links/eplan/2d9e6e6eb2-7eb6-4dac-b75b-b022b77618d7](http://www.andritz.com/de/hydro/boyabat/links/eplan/2d9e6e6eb2-7eb6-4dac-b75b-b022b77618d7)

**Signal Auxiliary_Rack/CPU_1/Channel_1/Pin_3 (389459... has been deleted.**

- Type: digital
- Text: 400 VAC Main distribution. busbar A. undervoltage
- KKS: G-MKA20-CL005-501

**Further Information**

- [http://www.andritz.com/de/hydro/boyabat/links/eplan/38945975-a91b-46d6-81de-d3a2119d2967](http://www.andritz.com/de/hydro/boyabat/links/eplan/38945975-a91b-46d6-81de-d3a2119d2967)

Data Source:
Project Role Concept

Data source:
Engineering Database
Process Measurement and Analysis: 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.

Virtual common data model

<table>
<thead>
<tr>
<th>Tool A Data Extract</th>
<th>Tool B Data Extract</th>
</tr>
</thead>
<tbody>
<tr>
<td>Electrical Signal</td>
<td>Software Variable</td>
</tr>
</tbody>
</table>

Transformer

Engineering Data Base

| Signal | Electrical Signal | Software Variable |

Start

Signal List

Checkin

Signal comparison

similar

No Change

Change Identified

Override

Yes

Next Signal

No

End

- **Signal Changes**
  - Modified signals
  - New signals
  - Removed signals
  - Accepted / rejected signals

- Notification of changes to related stakeholders

- Events (E1 .. E10) enable process observation and project control

- Evaluation: pilot application based on historical data.
Process Measurement and Analysis: Feasibility Study Concept

- **Goal:**
  - Verification and validation of signal change management (process behavior)
  - Definition of project metrics, i.e., number of change per engineering phase / check-in sequence) for project monitoring and control.

- **Measurement Data & Metrics**
  - Events (E1 … E10)
  - Definition of Product and Project Metrics based on signal changes.

- **Material:**
  - Real world project (hydro power plant) with three different signal lists in early phases of development (approx. 700 signals per list).

- **Process:**
  - Check-in of different signal lists
  - Capturing event data
  - Analysis of event data for process evaluation and determination of product metrics.
Process Measurement and Analysis: Results of the Feasibility Study

- Process Evaluation with ProM*

- Project monitoring and observation based on event data

Added-Value Contributions on three Levels

- **Added-value application examples**
  - Interact with project-level engineering knowledge and data.
  - Engineering Cockpit.
  - Use Case “Signal Deletion with Tickets”.
  - Efficient change conflict resolution.

- **Semantic integration on project level**
  - Project-level concepts.
  - Mapping to tool concepts.

- **Technical integration of tools**
  - Engineering tools.
  - Infrastructure, Security.
  - Application-specific components.
Prototype Implementation: Engineering Cockpit

In distributed (automation) engineering projects …

- Data sets of several engineering groups evolve concurrently, often **without project-wide version management and progress tracking**.

- Lead engineers and managers get a **clear picture only shortly before project milestones**, seeing risks unnecessarily late.

- Our prototype solution will provide engineers and managers with
  - a platform to organize and perform specific inter-domain and inter-tool tasks.
  - means to collaborate efficiently within the engineering team.
  - integrated data on project progress and risks as soon as the engineer groups check in their local data sets to allow adjustments early.
Prototype Implementation:
Engineering Cockpit – Management View
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.
Automation Service Bus (ASB)

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

- 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).