Overcoming the challenges of reuse in large process model repositories

presented by Barbara Weber

Parts of this research was funded by the Austrian Science Fund (FWF): P23699-N23.
Agenda

- Motivation
- Process of Process Modeling
- Supporting the Process of Process Modeling
  - Refactoring
  - Test driven Modeling
  - Literate Process Modeling
- Summary
quality of process models as a necessary precondition for reuse

MOTIVATION
Reuse of Common Process Knowledge

- Increasing adoption of process-aware information systems

- Emergence of large process model repositories often comprising a large number of related process fragments

- Reuse of common process knowledge to reduce process modeling and maintenance efforts
Quality Problems

- Error rates between 10% and 50% in industrial process model collections (Mendling 2009, Mendling et al. 2008)
  - Non intention-revealing or inconsistent naming (Mendling et al. 2010)
  - Redundant process fragments (Hallerbach et al. 2010)
  - Large and unnecessarily complex process models (Soto et al. 2008)
Quality Problems

- Error rates between 10% and 50% in industrial process model collections (Mendling 2009, Mendling et al. 2008)
  - Non intention-revealing or inconsistent naming (Mendling et al. 2010)
  - Redundant process fragments (Hallerbach et al. 2010)
  - Large and unnecessarily complex process models (Soto et al. 2008)
how does the process of creating, modifying, and reusing process models look like

THE PROCESS OF PROCESS MODELING
Process of Process Modeling (PPM)

• Iterative, highly flexible process
• Depends on individual modeler
• 3 successive phases

(Pinggera et al. 2010, Pinggera et al. 2011, Pinggera et al. 2012)
Process of Process Modeling

Comprehension

• Understand requirements
• Understand existing process model
• Understand components for reuse
• Chunking (Cant et al. 1995)
  – Understanding in chunks (group of information)
Process of Process Modeling

Comprehension

- Central Concept: Working Memory
  - Required by all conscious mental activities
  - Severely limited: 7 +/- 2 information „slots“ (Miller 1956)
  - Mental effort: utilization of working memory (Paas et al. 2003)
  - Overflow: rapid performance decrease! (Sweller 1988)
Process of Process Modeling

Comprehension

• **Abstraction**
  - hiding of irrelevant information (Parnas 1972)
  - supports human mind’s attention management (Larking and Simon 1987)

• **Split-attention effect** *(Sweller and Chandler 1994)*
  - occurs when information from several sources needs to be integrated
  - switching attention between models
Process of Process Modeling

Comprehension

• **External memory**
  o mechanism for reducing mental effort
  o Information storage outside the human cognitive system (e.g., pencil and paper or a blackboard)

• **Cognitive Trace**
  o Information taken from working memory and stored in an external memory (e.g., to mark, update, and highlight information)
Process of Process Modeling

Modeling

- comprehended chunks are formalized in process model
  - by creating new model elements
  - by integrating reusable components

- varying number of modeling steps
Process of Process Modeling
Reconciliation

• improve understandability
  – reorganize model (refactor)
  – utilize secondary notation, typographic cues

• facilitate next comprehension phase
fostering reusability of large process model repositories

REFACTORING
Process Model Refactoring
Reconciliation Support Fostering Reuse

• Improving model quality without changing the observable behavior of the model
## Typical Process Model Smells

<table>
<thead>
<tr>
<th>PMS1: Non-intention Revealing Naming of Activities / Process Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>PMS2: Contrived Complexity</td>
</tr>
<tr>
<td>PMS3: Redundant Process Fragment</td>
</tr>
<tr>
<td>PMS4: Large Process Models</td>
</tr>
<tr>
<td>PMS5: Lazy Process Models</td>
</tr>
<tr>
<td>PMS6: Unused Branches</td>
</tr>
<tr>
<td>PMS7: Frequently Occurring Instance Changes</td>
</tr>
<tr>
<td>PMS8: Frequently Occurring Variant Changes</td>
</tr>
</tbody>
</table>

(Weber and Reichert 2008, Weber et al. 2011)
Identification of Process Model Smells

Application of Refactoring Techniques
## Process Model Refactorings

<table>
<thead>
<tr>
<th>Refactor</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RF1: Rename Activity</td>
<td></td>
</tr>
<tr>
<td>RF2: Rename Process Schema</td>
<td></td>
</tr>
<tr>
<td>RF3: Substitute Process Fragment</td>
<td></td>
</tr>
<tr>
<td>RF4: Extract Process Fragment</td>
<td></td>
</tr>
<tr>
<td>RF5: Replace Process Fragment by Reference</td>
<td></td>
</tr>
<tr>
<td>RF6: Inline Process Fragment</td>
<td></td>
</tr>
<tr>
<td>RF7: Re-label Collection</td>
<td></td>
</tr>
<tr>
<td>RF8: Remove Redundancies</td>
<td></td>
</tr>
<tr>
<td>RF9: Generalize Variant Change</td>
<td></td>
</tr>
<tr>
<td>RF10: Remove Unused Branches</td>
<td></td>
</tr>
<tr>
<td>RF11: Pull Up Instance Change</td>
<td></td>
</tr>
</tbody>
</table>

(Weber and Reichert 2008, Weber et al. 2011)
Labeling of Process Models

• PMS1. Non intention revealing naming of activities / process models
  - Ambiguous or non intention revealing labels
  - Inconsistent use of labeling styles

(Weber et al. 2011)
Number of ways students name an activity in a process model

• Insights from a process modeling experiment with 113 students.

• The following sentence in the process description resulted into **84 different ways** for naming this particular activity:

  “Afterwards the scouting team attends games of the player they are interested in live in the football stadium.”

(Fahland 2012)
Inconsistent Names and Labeling Styles

• Repository with 70 process models from healthcare

• 16 out of 70 process models contained activities regarding the scheduling of medical procedures (e.g., surgeries, medical examinations, drug administration)

• Although activities had similar intentions, different labels and labeling styles were used
  o “Make appointment”, “appointment”, “schedule examination”, “fix day”, “agree on surgery date”, “plan”

(Weber et al. 2011)
Labeling Conventions + Refactoring of Labels

• Usage of labeling conventions and domain thesauri  
  (Becker et al. 2009)

• Technique for automatically refactoring from action-noun style to labels in verb-object style  
  (Leopold, Smirnov, and Mendling 2012)

• Activities with similar labels provide potential refactoring opportunities  
  (Dijkman et al. 2011)
Redundant Process Fragments
Candidates for Reuse

• PMS3: Redundant Process Fragments
  
  o Redundant process fragments can be commonly found in existing process models
   (Weber et al. 2011)
  
  o More than 560 clones in the SAP reference model
   (Dumas et al. 20xx)
Extracting Sub Processes

• Method for automatically detecting exact clones, which can be extracted to sub processes (Dumas et al. 20xx)

  ▪ Similar process fragments as potential refactoring opportunities (Dijkman et al. 2011)
    o Exact matches
    o Similar fragments, whereby some activities only appear in one fragment
    o Same activities, but different business objects
Extracting Sub Processes

• Method for automatically detecting exact clones, which can be extracted (Dumas et al. 20xx)

  - Similar process fragments as potential refactoring opportunities (Dijkman et al. 2011)
    - Exact matches
    - Similar fragments, whereby some activities only appear in one fragment
    - Same activities, but different business objects

What granularity should a process fragment have to be best suited for reuse?
Large Process Models are Difficult to Understand, Maintain, and to Reuse

• PMS4: Large Process Models

  o Process models with several hundred activities are not uncommon (Soto et al. 2008)

  o Large process models tend to comprise more formal flaws than smaller ones (Mendling et al. 2008)
Automatically Extracting Sub processes

• Method for automatic modularization of business process models (Reijers et al. 2011)

• Method for the automatic labeling of process models (Leopold et al. 2011)
Automatically Extracting Sub processes

• Method for automatic modularization of business process models (Reijers et al. 2011)

• Method for the automatic labeling of process models (Leopold et al. 2011)

Are such automatically created sub processes good candidates for reuse?
fostering understandability, maintainability, and reuse of declarative processes

TEST DRIVEN MODELING
Test Driven Modeling

Comprehension Support Fostering Reuse

• Is the model really doing what it is expected to do?
Interactions between Constraints Make Declarative Processes Difficult to Understand

Each execution of write response letter must be preceded by read reviews for revising paper

Must be first activity executed

Get acceptance must not be followed by Work on revision

Must be executed at least once

Complete paper writing must be executed before Format to instructions

Must be executed once

(Zugal et al. 2012b)
Interactions between Constraints Make Design Processes Difficult to Understand

Understanding declarative models requires the model reader to keep all constraints and their dependencies in his/her mind

=> No support for computational offloading
Interactions between Constraints Make Declarative Processes Difficult to Understand

Each execution of write response must be preceded by read reviews.

Get acceptance must not be followed by work on revision.

Must be first activity executed.

How can we ensure that the model is really doing what it is expected to do (especially when adapting it)?

Complete paper writing must be executed before Format to instructions.

Must be executed once.
Test Driven Modeling

- Central Concept
  - Test Cases: allows to specify behavior the process model must exhibit or prohibit
  - Test Case consist of
    - Execution Trace (1)
    - Set of assertions (2), (3), (4)

(Zugal et al. 2012a)
Test Driven Modeling Suite

(Zugal et al. 2012a)
Test Driven Modeling

- Makes information explicit that is only implicit in the process model, thus fostering understandability by supporting computational offloading

- Allows for testing execution traces in an automated way

- Test cases constitute an executable documentation

- Reduces mental effort, speeds up changes, and reduces errors (Zugal et al. 2011, Zugal et al. 2012c)
fostering understandability, maintainability and reuse by intervening text and model

LITERATE PROCESS MODELING
Literate Process Modeling
Comprehension Support Fostering Reuse

• What is the model doing and why have certain design decisions been taken?
Literate Process Modeling

- Supports domain experts and system analysts during model comprehension by flexibly interlinking textual descriptions and formal process model

- Fosters model understandability
- Provides a documentation of the process model
- Facilitates model comprehension as well as model changes by documenting design decisions

(Pinggera et al. 2012b)
Literate Process Modeling

(Pinggera et al. 2012b)
SUMMARY

wrap up
Conclusions

• Quality a necessary precondition for reuse
• Creating, maintaining, and reusing process fragments is a process in itself, which requires support
References


References

References


References


References


THANK YOU FOR YOUR ATTENTION