Role-based Adaptation of Structural Reference Models to Application Models

Status Talk, 30.11.2018

Supervisors:
- Prof. Dr. Susanne Strahringer
- Prof. Dr. Frank J. Furrer
- Prof. Dr. Uwe Aßmann
Reference Models for Business Knowledge

“Reference models are generic conceptual models that formalize state-of-the-art knowledge of a certain domain. Adaptation mechanisms can assist to tailor reference models to specific business or project needs.” [Becker2007]

**Structured / Object Orientation**
Languages: UML, SysML, MARTE, xUML, ER, ...
- Represent the unique composition of the enterprise’s system parts
- Necessary for implementation

→ **Composition** of the system

**Procedural Models**
Knowledge by processes.

**Chained / State Orientation**
Languages: BPMN, BPEL, Petri Nets, EPC, Activity Diagrams, ...
- Display the unique workflow execution of an enterprise system
- Frequently used as requirement specification

→ **Behavior** of the system
Structural Reference Modeling in Practice

1. **Start**: industry standard reference model
2. Enterprise specific requirements and workflows
3. Domain adaptation of the reference model
4. **End**: final application model (and implementation)

**Challenge 1**: reuse of the structure

**Challenge 2**: implement the specific business logic

**Structural Models**
Knowledge by *objects*. 

Icons: https://www.flaticon.com/authors/eucalyp
Adaptation Problems with Object-Orientation

Challenge 1
Reuse and tailoring of the structural RM.

Inflexibility
Structural characteristics may limit the application of the template RM

Adhere to the template?

Invasive Changes
The adaptation is not separated and future RM evolutions are hard to trace back

Challenge 2
Modeling towards existent business logic.

[Kühn2018]

Multitasking
Multiple entities associated for the same tasks

Multi-classification
An entity is classified with several features

State-dependency
An entity is dependent on its state
### Roles as a Solution

**Business Object**

```plaintext
:Person
name = "Peter"
taxId = 0493027940
marriedTo = "Silvie"
fatherTo = "Vanessa"
employeeOf = "Folgswagen"
heart = "heart12303"
knee = "knee23"
foodInStomach = "apple"
age = "grown-up"
```

**Adaptations**

```plaintext
:Person
name = "Peter"
taxId = 0493027940
<<core>>
mixedTo = "Silvie"
fatherTo = "Vanessa"
employeeOf = "Folgswagen"
<<roles>>
heart = "heart12303"
knee = "knee23"
foodInStomach = "apple"
<<intrinsic parts>>
age = "grown-up"
<<phases>>
```

---

**Diagrams:**
- Pizza
- PizzaBase
- PizzaTopping
- SingleMeal
- MixedItem
- PanPizza
- PanPizzaExtraCheese
- Calzone
- Burger
- Classic
- PanPizzaCheese
- Cheese
- Served
- Temperature
- Table

---

Role-based Reference Model Adaptation
Hendrik Schön
Status Talk, 30.11.2018
Ordering a pizza

Domain: Pizza Restaurant

Business Logic: Ordering a Pizza, ...

Role-based Reference Model Adaptation
Hendrik Schön
Status Talk, 30.11.2018

Slide 6
Research Hypothesis

*With role-based reference model adaptation, we ...*

1. ... combine **structure and behavior** of the model elements in a simple way, leading to more expressiveness.
2. ... provide a **structured and traceable** way of business object adaptation for industry standard reference models.
3. ... enable multiple viewpoints and adaptations of single model elements in different ways for more flexibility.
Research Questions

How can the use of roles help to better adapt structural reference models?

1. Roles: How can a role represent a business object in (conceptual) modeling?

2. Adaptation: How do we adapt a structural reference model in a systematic way?

3. Dynamics: How may we model the dynamics of a system in structural reference models?
Research Scopes

- My Ph.D. Topic
- Roles (the new modeling mechanism)
- Adaptation (use-case specification)
- Dynamics (behavior specification)

Roles and the Dynamic Behavior

Roles as a Tool for Adaptation

Behavior-aware Adaptation
State of the Art
Roles

The RoSI Roles

Roles as an Enterprise Modeling Concept

Designing Application Domain Models with Roles

Further Work

The RoSI Roles

• basically the RoSI subjects related to SWE
  • General semantics and syntax
  • CROM
  • SCROLL
  • RSQI
  • RSUM
  • ...

• known authors within this area
  • Steimann
  • Riehle
  • Guizzardi
  • Almeida
  • ...
Delegation: An important concept for the appropriate design of object models

• Shows the difference between aggregation and delegation
• States the delegation as must-have in conceptual modeling
• Role properties
  • Delegation = role object + role filler object
  • Role dispatches unknown messages to object
  • Role represents the objects services and state
• *Roles are classes with special relationship
• Limitations of role usage:
  • *Not every object may be an filler
    (must be of a specific “role” class kind)
  • *A role may only be used by a single filler
    (model integrity > flexibility)
  • Deep roles: yes, cyclic associations: no

Designing Application Domain Models with Roles

• “Task” as a collaboration of involved roles
• Roles represent business logic:
  • Conceptual modeling (UML)
  • RRC cards
  • Sequence diagram
• *Lifetime and runtime behavior of roles
• States four role characteristics
  • Responsibility-driven
  • Separation of concerns
  • Dynamic and flexible
  • Reusable and adaptable
• *Role “collaboration” limited to message flows sequences

Adaptation

Reference Model Adaptation Classification

On the Syntax of Reference Model Configuration

Configurable Workflow Models

Customization of Domain-Specific Reference Models for Data Warehouses

Utilizing domain models for application design and validation

Further Work

Reference Model Adaptation Classification

- **Configuration**
  - by selection
  - The application domain can be described fully in design time.

- **Instantiation**
  - by embedding
  - The application domain can be covered by a general framework.

- **Specialization**
  - by revising
  - The application domain can be covered by a core solution.

- **Aggregation**
  - by combination
  - The application domain can be described partly.

- **Analogy**
  - by creativity
  - The application domain can be described by certain patterns recurring in each application.

On the Syntax of Reference Model Configuration

- Based on C-EPC's, mapping to EPC's
  [Dreiling2005, Rosemann2007]

- Configurable elements
  - Functions
    (on, off, optional)
  - Connectors
    (OR, XOR, AND)
  - Requirements
    (if A = OFF then OR = AND)
  - Guidelines
    (if D = ON then E = ON)

- Syntactical validation via EPC markup language
  (“EMPL”, XML based)

Configurable Workflow Models

- C-YAWL = event workflow (EWF) graph, based on petri nets
- Process consists of:
  - **Tasks** (like petri net transitions)
  - **Conditions** (like petri net places)
- **Tasks** consists of:
  - Inputs/Outputs (constraints like AND, OR, ...)
  - Ports (variability point for specialization)
- Ports can be activated, blocked or hidden

Customization of Domain-Specific Reference Models for Data Warehouses

- “Reference Multidimensional Models” (RMDM)
- Adaptation method for data objects
- Adaptation by customizing/revising (add, remove)
  - Attributes
  - Measurement
  - Data Classes, Levels, Hierarchies
  - Dimensions
- SQL syntax for schema creation with respect to the customization
- No customization of workflows, business logic, ...

Utilizing domain models for application design and validation

- "Application Domain Modeling" (ADOM)
- Based on UML profiles
- Specifies three layers:
  - Language (specifies UML 2.0 as meta-model)
  - Application (models of particular systems)
  - Domain (specification of application families)
- Domain layer enforces constraints on the application layer
- Application model elements derived via UML stereotypes from domain model (however, they are type-safe)
- Possible combination with sequence diagram (that is, in fact, structure + behavior)

Dynamics

Towards an Ontology of Scenes and Situations

Modeling events as entities in object-oriented conceptual modeling languages

Temporal Modeling and ORM

Representing Temporal Information in UML

Further Work


Towards an Ontology of Scenes and Situations

- Semantic for “things that happen"
- Introduces *scene* and *situation* into the UFO ontology
  - *Scene* is an overall entity of “something”, that is composed by situations
  - *Situations* are manifestations of scenes and composite entities, characterized by its inner elements (state of the objects and their relations)
- *Occurrences* are events that influence a scene and characterizes a situation
- Only foundational (no method, implementation, ...)

Modeling events as entities in object-oriented conceptual modeling languages

- Events as UML profile stereotypes
  - Structural Events (change in population of something)
  - Domain Events (sum of structural events changes character of the domain)
  - Action Request Events (an event that enforces reaction)
    - explicit (initiated explicitly by users)
    - temporal (an event that occurs due the passing of time)
    - generated (occurs when condition C is fulfilled)

- Event conditions and effects as OCL statements (e.g., “paper.submissionDate = time”)

- Events are still typed UML classes (no “first-class citizen” temporal modeling construct)

- No implementable operations/attributes

Temporal Modeling and ORM

- Temporal information for structural models, esp. “Object-Role Modeling” (ORM)
- Uses and expands Allen’s operators [Allen1990] for describing temporal period information in OWL time
- Entities as temporal object type
  - Definitional (true by default)
  - Once-only
  - Repeatable
- Only implicit events
- Time only as information, no “real” object life cycle

Representing Temporal Information in UML

- Based on UML profiles (still UML classes)
- Temporal as stereotype for objects and relationships
  - Durability (constant, permanent, durable, instantaneous)
  - Frequency (single, intermittent = roles)
- Temporal constraints specified in OCL
  self.salaryAt(t_today) >= self.salaryAtOrBefore(t_today-1)
- Historical entity types consist of lifespans (for storing within the system)
- Temporal specification without events but objects (e.g., “sell” as object)

## Comparison

### Approaches for Reference Model Adaptation

<table>
<thead>
<tr>
<th>Approach</th>
<th>Basis</th>
<th>Mechanism Profile</th>
<th>Entities</th>
<th>Relationships</th>
<th>Support</th>
<th>Time</th>
<th>Business Logic</th>
<th>Object Life Cycle</th>
</tr>
</thead>
<tbody>
<tr>
<td>C-EPC [Recker2005]</td>
<td>Proc</td>
<td>■ / □ / □ / □ / □</td>
<td>■</td>
<td>■</td>
<td>■ / ■</td>
<td>□ / □</td>
<td>□ / ■</td>
<td>□</td>
</tr>
<tr>
<td>C-YAWL [Gottschalk2008]</td>
<td>Proc</td>
<td>■ / □ / □ / □ / □</td>
<td>■</td>
<td>■</td>
<td>□ / ■</td>
<td>□ / □</td>
<td>□ / ■</td>
<td>□</td>
</tr>
<tr>
<td>RMDM [Schütz2014]</td>
<td>Struct</td>
<td>■ / □ / □ / □ / □</td>
<td>■</td>
<td>□</td>
<td>□ / □</td>
<td>□ / □</td>
<td>□ / □</td>
<td>□</td>
</tr>
<tr>
<td>ADOM-UML [Reinhartz2009]</td>
<td>Struct</td>
<td>□ / □ / □ / □ / □</td>
<td>■</td>
<td>■</td>
<td>□ / ■</td>
<td>□ / □</td>
<td>□ / □</td>
<td>□</td>
</tr>
<tr>
<td>UFO Scenes [Almeida2018]</td>
<td>Struct</td>
<td>□ / □ / □ / □ / □</td>
<td>■</td>
<td>■</td>
<td>□ / □</td>
<td>□ / □</td>
<td>□ / □</td>
<td>□</td>
</tr>
<tr>
<td>ORM [Halpin2008]</td>
<td>Struct</td>
<td>□ / □ / □ / □ / □</td>
<td>■</td>
<td>■</td>
<td>□ / □</td>
<td>□ / □</td>
<td>□ / □</td>
<td>□</td>
</tr>
<tr>
<td>Temporal UML [Cabot2003]</td>
<td>Struct</td>
<td>□ / □ / □ / □ / □</td>
<td>■</td>
<td>■</td>
<td>□ / ■</td>
<td>□ / □</td>
<td>□ / □</td>
<td>□</td>
</tr>
<tr>
<td>Our Target</td>
<td>Struct</td>
<td>■ / □ / □ / □ / □</td>
<td>■</td>
<td>■</td>
<td>■ / ■</td>
<td>□ / □</td>
<td>□ / □</td>
<td>□</td>
</tr>
</tbody>
</table>

- ■ supported
- □ partially supported
- □ not included
Role-based Reference Modeling
Research Approach

Language Specification

Method Specification

Method

- Business Role-Object Specification (BROS)
  - Entities
  - Relationships
  - Constraints
  - Properties

- Reference Model Adaptation with Roles (RAWR)
  - Instructions
  - Rules
  - Guidelines
  - Patterns

Tools

- Graphical Modeling Editor
  - Create, store, import, and export models
  - Method support
  - User support by common IDE features (error highlighting, auto completion, ...)

Technische Universität Dresden

Role-based Reference Model Adaptation
Hendrik Schön
Status Talk, 30.11.2018

Slide 28
Language Specification

Business Role-Object Specification
BROS

- Modeling of a systems structure
- Roles as participants in business logic [Colman 2006]
- BROS meta-model influenced by CROM meta-model [Kühn 2014]
- Provides a concept for features/applications

“All the world’s a stage, and all the men and women merely players: they have their exits and their entrances; and one man in his time plays many parts, his acts being seven ages.”

– William Shakespeare

Language Specification

Method Specification

Business Object(s)

standardized, prescribed

[Colman2006]

BROS meta-model influenced by CROM meta-model [Kühn 2014]

Provides a concept for features/applications

“...”

Compartments

Naturals (Objects)

Events: modeling construct for controlling the temporal behavior of roles

Roles

Scenes: temporal collaboration of roles and events, related to common business logic
Language Specification

Domain Model

BROS Model

Process Model

Language Specification

Method Specification
Ordering another pizza

Domain: Pizza Restaurant

Business Logic: Ordering a Pizza, ...
Ordering another pizza

main model elements:
- BROS model
- Scenes
- Roles
- Players
- Events
- Global- and Sub-Scenes
Method Specification I

Reference Model Adaptation with Roles

RAWR

- User manual for reference model adaptation
- Leads to defined adaptation processes

Reference Model

Business Objects

Application Model

Roles

Language Specification

Method Specification

(work in progress) standardized, prescribed user made, adapted

feedback

Method Specification II

Method Specification III

Method Specification IV

Method Specification V
Method Specification II

Reference Model Adaptation with Roles

RAWR

- Instruction set
  - Determined instructions
  - Set of rules per instruction
  - Set of guidelines per instruction
- Patterns
  - Solutions to common modeling issues
  - Based on role-event usage

<table>
<thead>
<tr>
<th>Instr.</th>
<th>Rules</th>
<th>Guidelines</th>
</tr>
</thead>
<tbody>
<tr>
<td>(I1) Select a appropriate view of the reference model.</td>
<td>(R.I1-1) Views contain a subset of reference model objects and their relationships. (R.I1-2) Views cannot be empty. (R.I1-3) Views have an identifier, author, creation date, and modif. date. They may have attributes or operations as global variables. (R.I1-4) Views can only contain each reference model object once. (R.I1-5) Relationships of a view’s objects with a target outside the view are not included into the view.</td>
<td>(G.I1-1) A view should encapsulate suitable related objects and relationships. (G.I1-2) A view represents the modelers intention. (G.I1-3) Views are not self-contained. If they use already adapted objects of other views, this should be synchronized. (G.I1-4) Scenes may be exclusive to a view, however, the scene is part of the general adaptation.</td>
</tr>
<tr>
<td>(I2) Create scenes and sub-scenes for the adaptation.</td>
<td>(R.I1-1) ...</td>
<td>(G.I1-1) ...</td>
</tr>
</tbody>
</table>
Method Evaluation

Adaptation Use Case
- Cooperation with cloud service provider
- Using “real world” reference model
- Adaptation towards a new requirement:
  → “Partnership” program
  → 10% off when recruiting a new customer
  → Lasts 1 year or until contract termination
Approach Overview

Language Specification

Method Specification

BROS

Business Role-Object Specification

RAWR

Reference Model Adaptation with Roles

Role-based Reference Model Adaptation

Language Specification

Method Specification

BROS

Business Role-Object Specification

RAWR

Reference Model Adaptation with Roles

Instr. | Rules | Guidelines
--- | --- | ---
(I1) Select a appropriate view of the reference model. | (G.II-1) A view should encapsulate suitable related objects and relationships. | (G.II-2) A view represents the modelers intension. |
(R.II-1) Views contain a subset of reference model objects and their relationships. | (G.II-3) Views are not self-contained. If they use already adapted objects of other views, this should be synchronized. | |
(R.II-2) Views cannot be empty. | (G.II-4) Scenes may be exclusive to a view, however, the scene is part of the general adaptation. | |
(R.II-3) Views have an identifier, author, creation date, and modif. date. They may have attributes or operations as global variables. | (G.II-5) Relationships of a view’s objects with a target outside the view are not included into the view. | |
(R.II-4) Views can only contain each reference model object once. | (G.II-6) … | |
(R.II-5) Relationships of a view’s objects with a target outside the view are not included into the view. | (G.II-7) … | |
Current Ph.D. Progress

Role-based Adaptation of Domain Reference Models: Suggestion of a Novel Approach.

Revision in review. (2018)


Additional papers are done/planned in cooperation with C. Werner, T. Kühn, and M. Wutzler.
Open Issues and Outlook

Future Main Work

- Establish the BROS specification
  - SCROLL
  - CROM
  - RoleDiSCo
  - LTL (?)
- Finish the RAWR method
  - Instructions, rules and guidelines
  - Evaluation
- Further tool development (*framed.io*)

Timeline of 2019

<table>
<thead>
<tr>
<th>Jan</th>
<th>RAWR Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feb</td>
<td>(instruction set, evaluation)</td>
</tr>
<tr>
<td>Mar</td>
<td></td>
</tr>
<tr>
<td>Apr</td>
<td>Stay Abroad at University of Stockholm (<em>Jelena Zdravkovic</em>)</td>
</tr>
<tr>
<td>May</td>
<td>Department of Computer Science, Enterprise Modeling Group</td>
</tr>
<tr>
<td></td>
<td>(work together, finish method, further evaluation)</td>
</tr>
<tr>
<td>Jun</td>
<td></td>
</tr>
<tr>
<td>Jul</td>
<td>Writing the thesis</td>
</tr>
<tr>
<td>Aug</td>
<td>(side work: application/evaluation of BROS/RAWR)</td>
</tr>
<tr>
<td>Sep</td>
<td></td>
</tr>
<tr>
<td>Oct</td>
<td></td>
</tr>
<tr>
<td>Nov</td>
<td></td>
</tr>
<tr>
<td>Dec</td>
<td></td>
</tr>
</tbody>
</table>
“The reference model becomes a lessons-learned repository for all to share.”

Edward Averill
Reference Models and Standards, 1994

Thank you for your attention.
References


