Constraint management in engineering of complex information systems


marie-noelle.terrasse@u-bourgogne.fr
A gap between the semantical and operational organizations:

- Semantical units of knowledge
  small-grain, highly inter-related
  abstract constraints, i.e., compliance, dependencies
A gap between the semantical and operational organizations:

- Semantical units of knowledge
  small-grain, highly inter-related
  **abstract constraints**, e.g., compliance, dependencies

- Operational units of resources
  coarse-grain, automatically combined
  **constraints at various abstraction levels**
A gap between the semantical and operational organizations:

- Semantical units of knowledge
  - small-grain, highly inter-related
  - abstract constraints
- Operational units of resources
  - coarse-grain, automatically combined
  - constraints at various abstraction levels

Engineering processes:

- “ETL-like” use of resources:
  - E selection of relevant resources
  - T definition of organization criteria
  - L resource integration
- MDE-approaches
  - models at various abstraction levels

Constraints and architectures of models?
## Towards a typology of constraints

<table>
<thead>
<tr>
<th>Modeling Paradigm</th>
<th>Constraints on open-world and close-world assumptions</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modeling Language</td>
<td>Constraints on multiplicities of multiple associations on one class (UML/OCL)</td>
<td>Context facet-object</td>
</tr>
<tr>
<td></td>
<td></td>
<td><code>inv composition: not (self.contains1-&gt;isNotEmpty() and self.contains2-&gt;isNotEmpty())</code></td>
</tr>
<tr>
<td>Business Context</td>
<td>Functional CNIL-related constraints in France</td>
<td>SecureUML: <code>&lt;&lt;secureuml.permission&gt;&gt;</code>  <code>&lt;&lt;secureuml.role&gt;&gt;</code></td>
</tr>
<tr>
<td></td>
<td>Non-functional constraints due to security requirements</td>
<td>Mandatory use of the Uniprot accession numbers</td>
</tr>
<tr>
<td>Application Context</td>
<td>Functional multibase search for a protein sequence</td>
<td>Forbidden use of XHTML features</td>
</tr>
<tr>
<td></td>
<td>Non-functional an application’s web-site must have wide-scope</td>
<td></td>
</tr>
</tbody>
</table>

Abstraction versus finding, formulating, validating constraints.
Finding and formulating constraints

- Ontology-based approaches:
  “finding” is an incremental process
  uniform “formulation” carried out at all ontological levels

- Model-based approaches:
  “finding” with **semi-automatic constraint generators**
    - Costal, 2008
      typology of constraints (e.g., path-based, value-based)
      typology of operations (e.g., on instances, on relations)
    - model-checking
      SCR (Heitmeyer & al.), GME (Nordstrom & al.)
  uniform “formulation” of **structural constraints** (OCL)
  “formulation” of **behavioral constraints**
    - OCL *pre*- and *post*- conditions
    - OCL *body* clauses
    - Action Semantics Languages and code snippets
      e.g., EP Kelsen, 2007, ASOQ Hausten, 2004
Polymorphous constraints
a 3D-space (construct, diagram, abstraction level)

- Pre-condition (class diagram)
  context faceted-object::initialization()
  pre: self.contains1->notEmpty()

- Guarding condition (statechart)
  self.contains1->notEmpty()/initialization()

- Stereotypes
  class: facet, faceted, mandatory-facet
  association: facet-def
  operation: faceted-operation
Constraint verification

- **a posteriori**
  - check clauses (relational databases)
  - symbolic model checking (on UML models)

- **step-by-step**
  - a posteriori
  - proof-based

- **a priori**
  - DSL
  - metamodelling

- **OWL engines**
  - on A-boxes
    - "security" is an NF property
  - on T-boxes
    - all NF-properties are quantifiable

- **OCL checkers**
  - SAT-based
    - Kreiger, 2008

- **high-level ontologies**
  - BFO and OBO-RO
An illustrative exemple: XenOnt

an image

and its metadata:
- project structure
- scientific objective
- experimental protocol

FuGE (Common package)

expert knowledge

BFO and OBO-RO

pattern

FBbi
XAD

consensual resources
for annotation

- annotation A1
- annotation A2
- annotation A3
- annotation A4
The XenOnt’s constraints

1. model-level (model transf.)
   - OWL 1.1
   - BFO

2. metamodel and model (use of ODM stereotypes)
   - UML 2.0 & OCL 2.0
   - FuGE
   - FuGE

3. metamodel and model (use of stereotypes)
   - relational DBMS
   - RacerPro

4. instance-level (use of ontologies)
   - RacerPro

5. model-level (model transf.)
   - Protégé

6. metamodel
   - Protégé

7. model-level (model transf.)
   - Protégé

8. metamodel and model
   - Protégé

software

modeling languages, domain standards and ontologies

OBO

XAO

FBbi

GO

Terrasse & al.  Constraint management
Conclusion

• A 3D-space for constraint management in Model Driven Engineering
  combination of formal methods Clarke, 96 Lamsweerde, 00
  availability of relevant frameworks

• A major issue
  constraint traceability
    - while changing abstraction levels
    - during model transformations

• A DRE-perspective
  early constraint verification
  high-level ontologies, metamodels and DSLs
  enforcing the use of constructs that embed constraints
  might be easier in domain-specific environments