SiLift: Extending EMF Compare with an Operational View on Model Differences

Timo Kehrer
Software Engineering Group
University of Siegen

Eclipse DemoCamp
Stuttgart, 17.07.2013

Semantic Lifting of Model Differences

http://pi.informatik.uni-siegen.de/Projekte/SiLift
Outline

Introduction

The SiLift Approach

End-user Tool Environment

Summary
Context: Model-driven Development
Model evolution

rev. 1.1

```
1
Company
  + employer

worksFor

* Person
  + employee

  Developer
    + name : String

  Manager
    + name : String
```
Model evolution

rev. 1.1

*Person

+ employee

Company

+ employer

worksFor

1

Developer

+ name : String

Manager

+ name : String

Design Decision:
Restrict association navigability

rev. 1.2

*Person

+ employee

Company

+ employer

worksFor

1

Developer

+ name : String

Manager

+ name : String
Model evolution

rev. 1.1

Design Decision: Restrict association navigability

rev. 1.2

Refactoring: Pull up attribute

rev. 1.3

T. Kehrer: High-level Versioning of EMF Models with SiLift
Model versions and variants

1.1

1.2

1.3

1.2.2.1
3-Way model merging

1.1

1.2 base model

1.3 changed version A

1.2.2.1 changed version B
Model patching

1.1

1.2

1.3

1.2.2.1

propagate changes

origin model

changed model

target model
Basic challenge: Model comparison

T. Kehrer: High-level Versioning of EMF Models with SiLift
What textual difference tools report...
The processing pipeline of **EMF Compare**
Part 1: Matching

**Generic** Matchers, e.g.
- XMI ID
- Functional ID
- Content matching strategy

**Custom** Matching Engine
The processing pipeline of EMF Compare Part 2: Differencing

1. **Generic** solutions operating on the models' ASG „low-level“ changes

Adaption to custom high-level edit operations
Outline

Introduction

The SiLift Approach

End-user Tool Environment

Summary
Overall goal: Lift model versioning to the abstraction level of edit operations

High-level changes: Complex editing commands

Low-level model changes
Basic approach

Match → Difference Derivation → Low-level Changes → Edit Operation Detection → Edit Operations → Conflict & Dependency Analysis
Sample edit rule: „pullUpAttribute“ (theoretical foundation)

<table>
<thead>
<tr>
<th>pullUpAttribute(a)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( Sup )</td>
</tr>
<tr>
<td>( Sub )</td>
</tr>
<tr>
<td>( Sub' )</td>
</tr>
<tr>
<td>( a : dt )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LHS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RHS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

\[ \text{LHS} \rightarrow \text{RHS} \]
Sample edit rule: „pullUpAttribute“ (implementation in EMF Henshin)
Outline

Introduction

The SiLift Approach

End-user Tool Environment

Summary
Difference Viewer

- Resource Set
- Platform: `/resource/org.sidiff.uml2.examples/example/
- Semantic Change Set: "Restrict Navigability"
- Semantic Change Set: "Pull Up Attribute"

```
Company  worksFor [*] Person
  + employer  + employee

Developer
  + name : String

Manager
  + name : String

Company  worksFor [*] Person
  + employer  + employee

Developer

Manager
```

T. Kehrer: High-level Versioning of EMF Models with SiLift
Consistency-preserving editing of patches

base model $v_0$ → $v_1$ changed model

T. Kehrer: High-level Versioning of EMF Models with SiLift
Consistency-preserving editing of patches

base model $v_0$ → $v_1$ changed model

Diagram showing changes to a model from version $v_0$ to $v_1$.
Consistency-preserving editing of patches
Controlled application of model patches

T. Kehrer: High-level Versioning of EMF Models with SiLift
Controller application of model patches

Retrieval of operation context
Controlled application of model patches

Operation execution log

T. Kehrer: High-level Versioning of EMF Models with SiLift
Controlled application of model patches

Modifyable Target model
Controlled application of model patches

Result of the patch application
Outline

Introduction

The SiLift Approach

End-user Tool Environment

Summary
Summary

- Unit of change; high-level edit operations
- Formal foundation; graph transformation concepts
- Support for complex edit operations
- High-level conflict and dependency detection
- Consistency-preserving patching
Further information

- **SiLift project web site:**
  http://pi.informatik.uni-siegen.de/Projekte/SiLift


