EMF Diff / Merge

an engine for syntactically consistent merged models

Olivier Constant
Thales Global Services

EclipseCon Europe 2012, Modeling Symposium
Ludwigsburg, October 23
Background – Merging Models

Compare

Match

Model 1

Model 2

Model 1

Model 2

Model 1

Model 2 (merged)

Model 1

Model 2

Model 1

Model 2

Compare

Match

Diff

Merge
Why Merging Models?

A recurrent need

- Version control
- Iterative model transformation
- Model refactoring
- Bridge between model-based tools
- Others …

⇒ « Merge » as a fundamental feature when developing model-based tools

- But… that can be hazardous
Problem

An erroneous merge can break a model
- A difference is local to a model element
- A model is globally constrained by its metamodel
  ➞ Some differences are semantically related and should be merged together

What can be done?
- If metamodel is known, develop specific merging rules
- But what if: metamodel evolves, is unknown, or there are too many metamodels? Can we provide **minimal guarantees**?
Principle of EMF Diff/Merge

Initialise comparison with \{ Match, Diff, Merge \} policies

Forbid contradictions
Enforce consistency (syntactic)

Differences actually merged

Consistency Rules (metamodel)
The EMF Diff/Merge Project

Origin at Thales Global Services

◆ Spring 2010: the need is becoming too strong for us
◆ Summer 2010: definition of a conceptual basis
◆ Autumn 2010: start of prototyping phase
◆ Since then: operational usage for certain scenarios, integration into higher-level tools

Now at Eclipse

◆ Useful to us ➔ certainly useful to others
◆ Spring–Summer 2012: project creation (EMF sub-project)
  ◆ In the context of the AGeSys project (French "System@tic" ICT cluster)
◆ Intent: sound basis for helping develop higher-level tools

◆ Now available! Come and try it 😊
  http://wiki.eclipse.org/EMF_DiffMerge
Typical Usage

Programmatic

```java
IComparison c = new EComparisonImpl(scope1, scope2);
c.compute(matchPolicy, diffPolicy, mergePolicy, progressMonitor);
c.merge(differenceSelector, progressMonitor);
```

Contributive (sample GUI)

- **Definition of selectable comparison methods**
- **Comparison method =**
  - Applicability conditions
  - Model scope definition
  - Match, Diff, Merge policies

![EMF Diff/Merge](image)

**Define the comparison operation to perform**

Define the comparison method and the role played by each model in the comparison.

- **Roles**
  - Left: [GraphExample/Graph/1.elements](#)
  - Right: [GraphExample/Graph/2.elements](#)

- **Comparison method**
  - Matching by ID, support for GMF scope
  - Configure...