

# EMF Diff / Merge

an engine for syntactically consistent merged models

Olivier Constant  
Thales Global Services

EclipseCon Europe 2012, Modeling Symposium  
Ludwigsburg, October 23

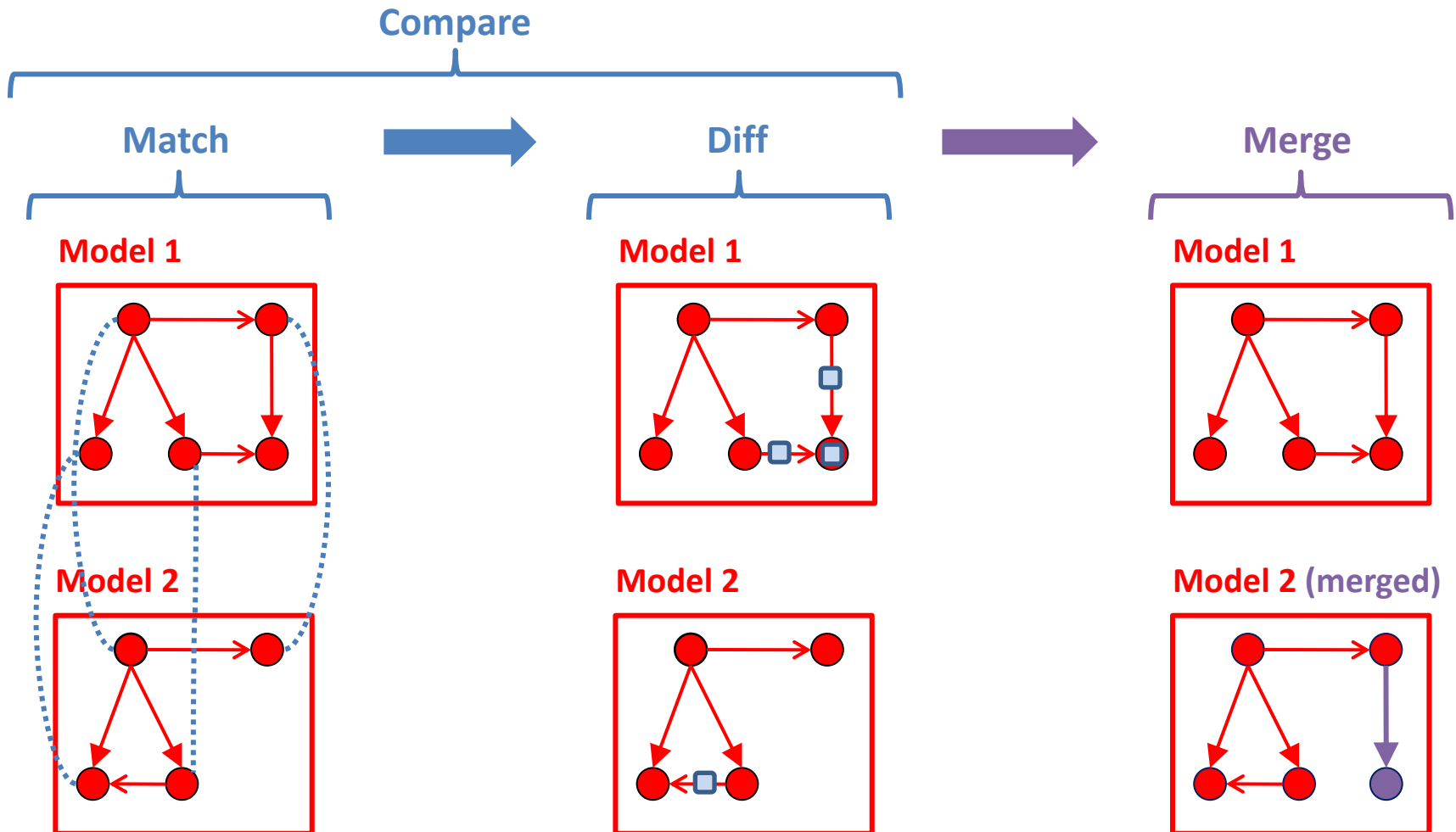
INFORMATION SYSTEMS  
HUMAN RESOURCES  
PURCHASING  
COMMUNICATION  
ENGINEERING  
TRANSFORMATION  
REAL ESTATE & FM

**Thales Global Services**  
the essence of expertise

OPEN

**THALES**

# Background – Merging Models



# Why Merging Models?

## A recurrent need

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

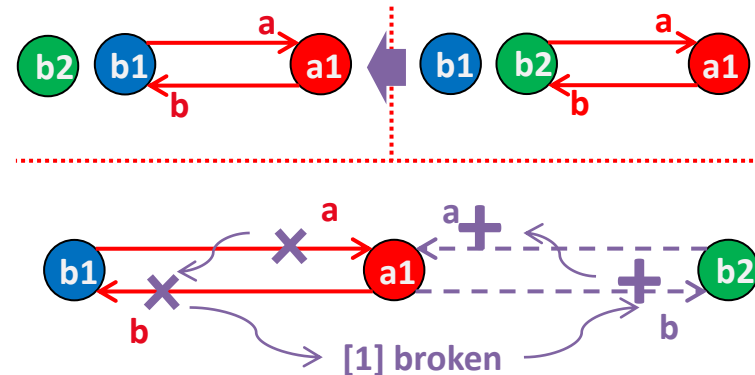
➔ « Merge » as a fundamental feature when **developing model-based tools**

- ◆ But... that can be hazardous

# Problem

## An erroneous merge can break a model

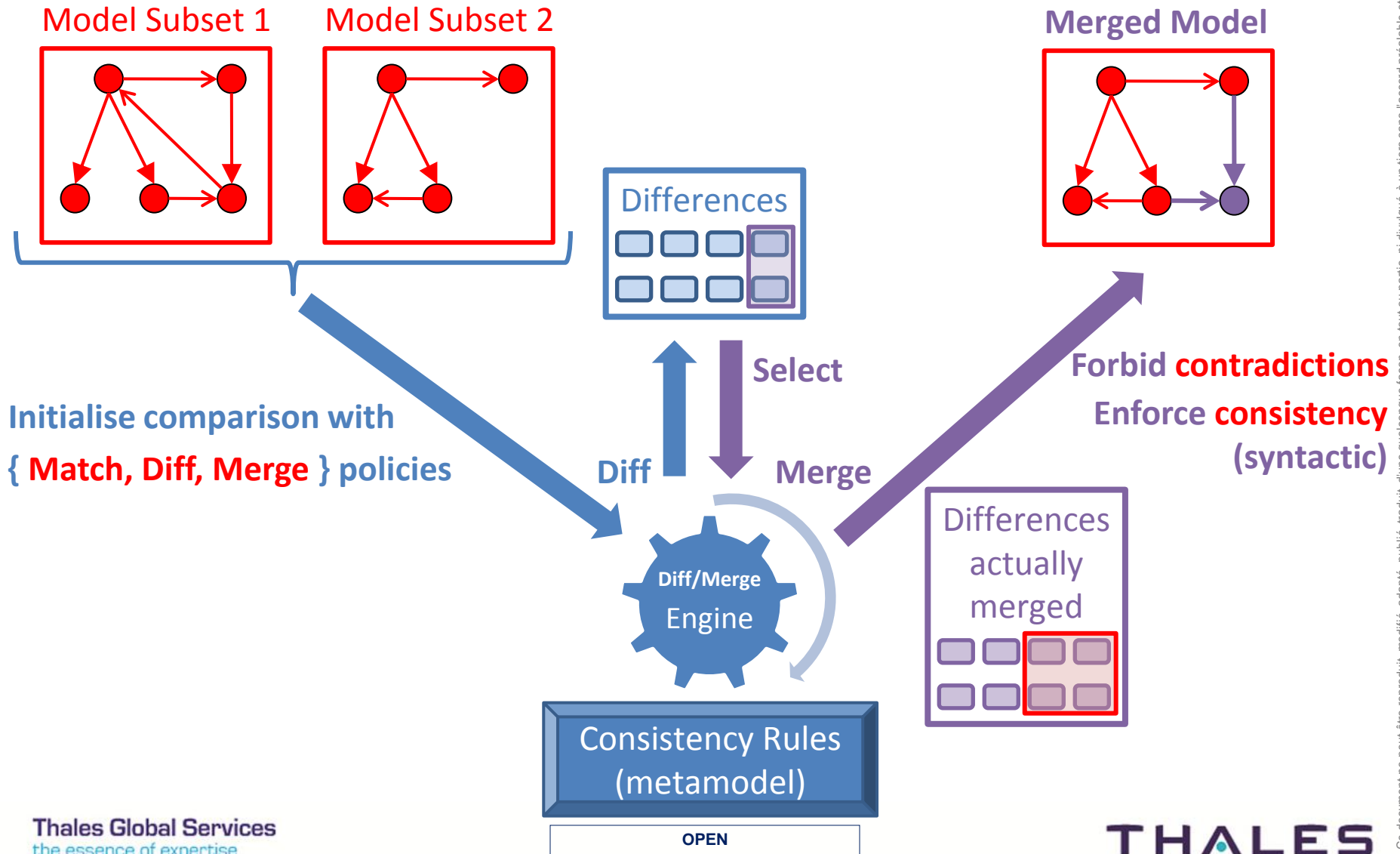
- ◆ A difference is local to a model element
- ◆ A model is globally constrained by its metamodel
- ➔ Some differences are semantically related and should be merged together



## 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



# The EMF Diff/Merge Project

## Origin at Thales Global Services

- ◆ Spring 2010: the need is becoming too strong for us
- ◆ Summer 2010: development of an initial **diff/merge engine**
- ◆ Autumn 2010: start of **operational usage** in Thales divisions
- ◆ Since then: more developments, integration into higher-level tools, usage in critical projects

## Now at Eclipse

- ◆ Useful to us → certainly useful to others
- ◆ Spring–Summer 2012: project creation (EMF sub-project)
- ◆ Intent: sound basis for helping develop higher-level tools
  
- ◆ Now available! Come and try it 😊  
[http://wiki.eclipse.org/EMF\\_DiffMerge](http://wiki.eclipse.org/EMF_DiffMerge)

# Demo

**EMF Diff/Merge**

Define the comparison operation to perform

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

**Roles**

Left: GraphExample/GraphV1.elements

Right: GraphExample/GraphV2.elements

**Comparison method**

Matching by ID, support for GMF scope

Configure...

Finish Cancel

**Initializing Compare Editor**

EMF Diff/Merge

Registering and matching elements

Always run in background

Run in Background Cancel Details >>

**Compare**

Synthesis

- Root (8)
  - Edge N3-to-NB-Container
    - Node N2 (4)
      - Node N221 (2)
        - Edge N221-to-N3
      - Node N3 (2)
      - Referencing Node ToN221

Details

**Merge Operation**

The merge operation will have the following impact on the model on the left.

**Required changes**

- Node N221
  - Deletion

**Implied changes**

- Edge N221-to-N3
  - Deletion
- Node N3
  - Reference 'incoming': deletion of Edge N221-to-N3
- Referencing Node ToN221
  - Deletion

OK Cancel

# Typical Usage

## Programmatic

```
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

