

EMF Diff / Merge

Thales Global Services

March 26th, 2012

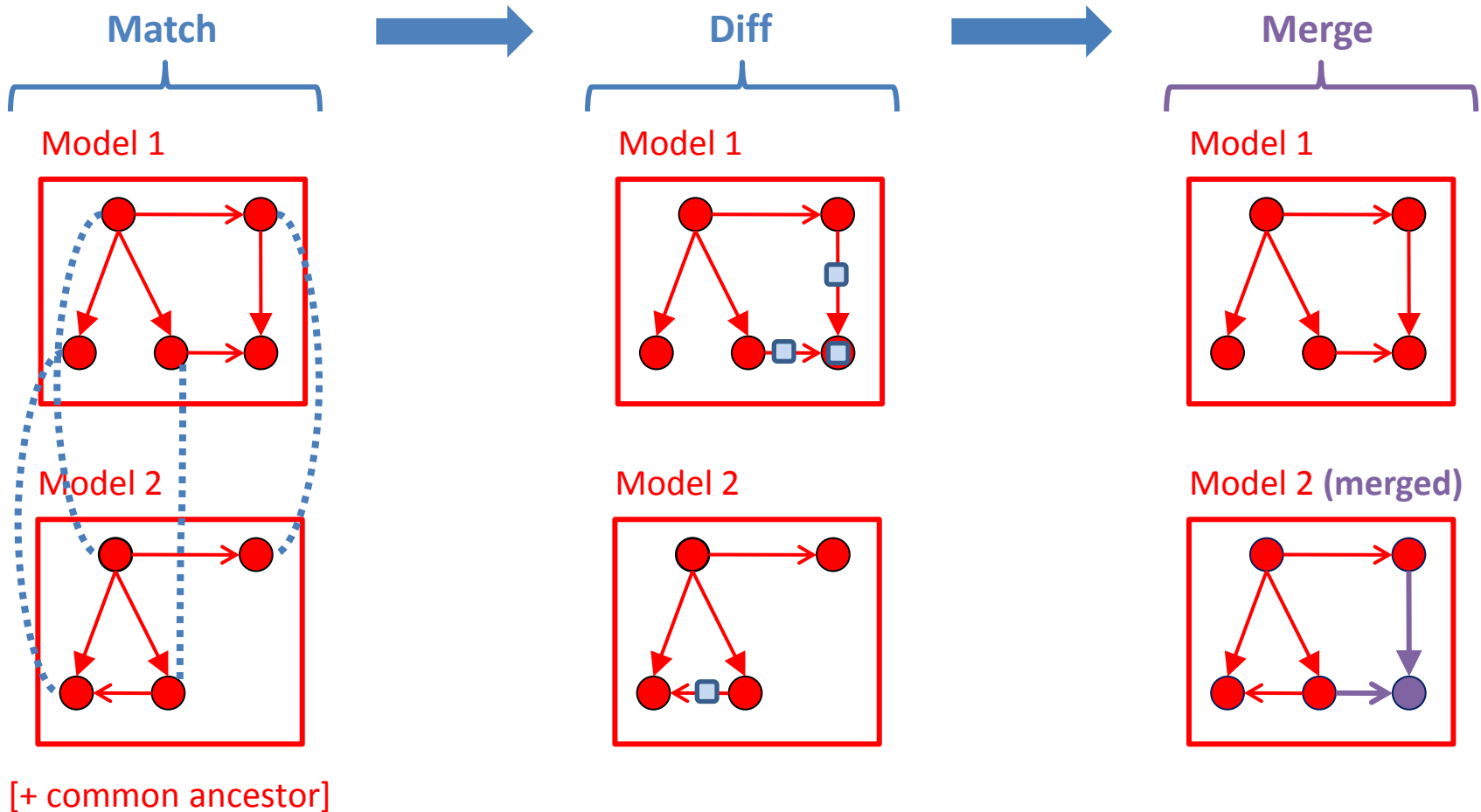
INFORMATION SYSTEMS
HUMAN RESOURCES
PURCHASING
COMMUNICATION
ENGINEERING
TRANSFORMATION
REAL ESTATE & FM

Thales Global Services
the essence of expertise

OPEN

THALES

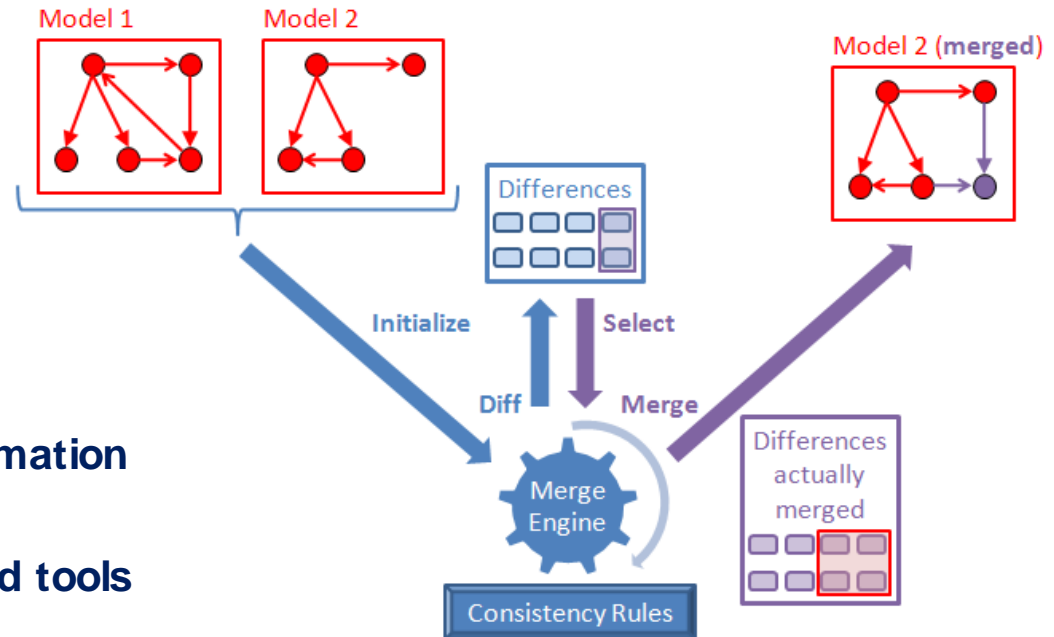
Background: merging models



A more specific need

Transforming models by merge while enforcing properties

- ◆ Preserve conformance to metamodel
 - ... or not, according to *consistency rules* and user-defined *policies*
- ◆ Merge in any order without consequences
- ◆ Merge parts of the same model



Typical use cases

- ◆ Assisted version control
- ◆ Incremental model transformation
- ◆ Model refactoring
- ◆ Bridge between model-based tools
- ◆ Others ...

The EM F Diff / Merge proposal

History

- ◆ **Diff/ Merge engine developed by Thales/ TGS for operational needs**
- ◆ **Used since Sept. 2010 in industrial projects**
 - Integrated into other tools
 - Features: version control, model refactoring, incremental transformations
- ◆ **Engine considered mature according to our criteria**

Present times

- ◆ **Idea: tool solves recurring problems → useful to other tools?**
 - E.g., CDO, ...
- ◆ **Proposal submission process underway: new sub-project under EM FT**
 - Scope: reusable engine and GUI components, matching by unique identifiers
 - Started in the context of the AGeSys project (French "System @tic" ICT cluster)
 - Contact: olivier.constant@thalesgroup.com

Typical usage

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

Demo

The screenshot displays a software development environment with a 'Compare' window and a 'Merge Operation' dialog box.

Compare Window:

- Synthesis:** Shows a tree structure with 'Root (3)' containing 'Node B (1)', 'Node C (1)', and 'Arrow Z (1)'.
- Example/03. Integrity-M2Violation/:** Shows a tree structure with 'Root' containing 'Node A', 'Node B', 'Node C', and 'Arrow Z'.
- Example/03. Integrity-M2Violation/t:** Shows a tree structure with 'Root' containing 'Node A', 'Node B', 'Node C', and 'Arrow Z'.
- Details:** The left pane shows 'incoming'. The right pane shows 'Arrow Z (in Root)'.

Merge Operation Dialog:

The dialog box is titled 'Merge Operation' and contains the following information:

- Message:** The merge operation will have the following impact on the model on the left.
- Required changes:**
 - Node C
 - Reference 'incoming': addition of Arrow Z
- Implied changes:**
 - Arrow Z
 - Reference 'target': addition of Node C
 - Reference 'target': deletion of Node B
 - Node B
 - Reference 'incoming': deletion of Arrow Z

Buttons: OK, Cancel