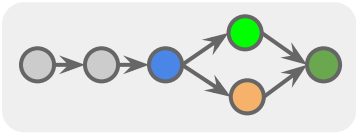
Tailor-made model comparison: How to customize EMF Compare for your modeling language

Philip Langer <planger@eclipsesource.com>
Laurent Delaigue <laurent.delaigue@obeo.fr>
Maximilian Koegel <mkoegel@eclipsesource.com>

EMF Compare

- Comparing and merging EMF-based models
  - Obtaining differences among model versions
    - Two-way differencing
    - Three-way differencing
  - Merging concurrently modified model versions
  - Merge viewers
    - Showing differences
    - Merging differences
    - Resolving conflicts
  - Integration with Eclipse team providers, such as EGit
  - Ships with dedicated support for Papyrus UML and GMF



EMF Compare + EGit =

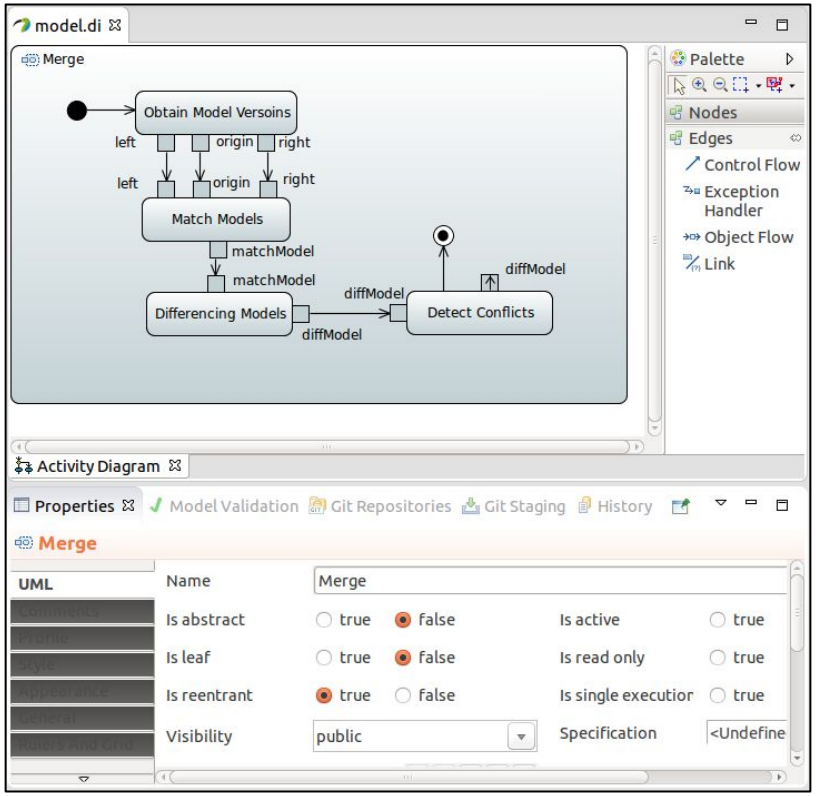
papyrus
www.eclipse.org/papyrus

EclipseSource

beo Model Driven Company

EMF Compare

The way how you model...

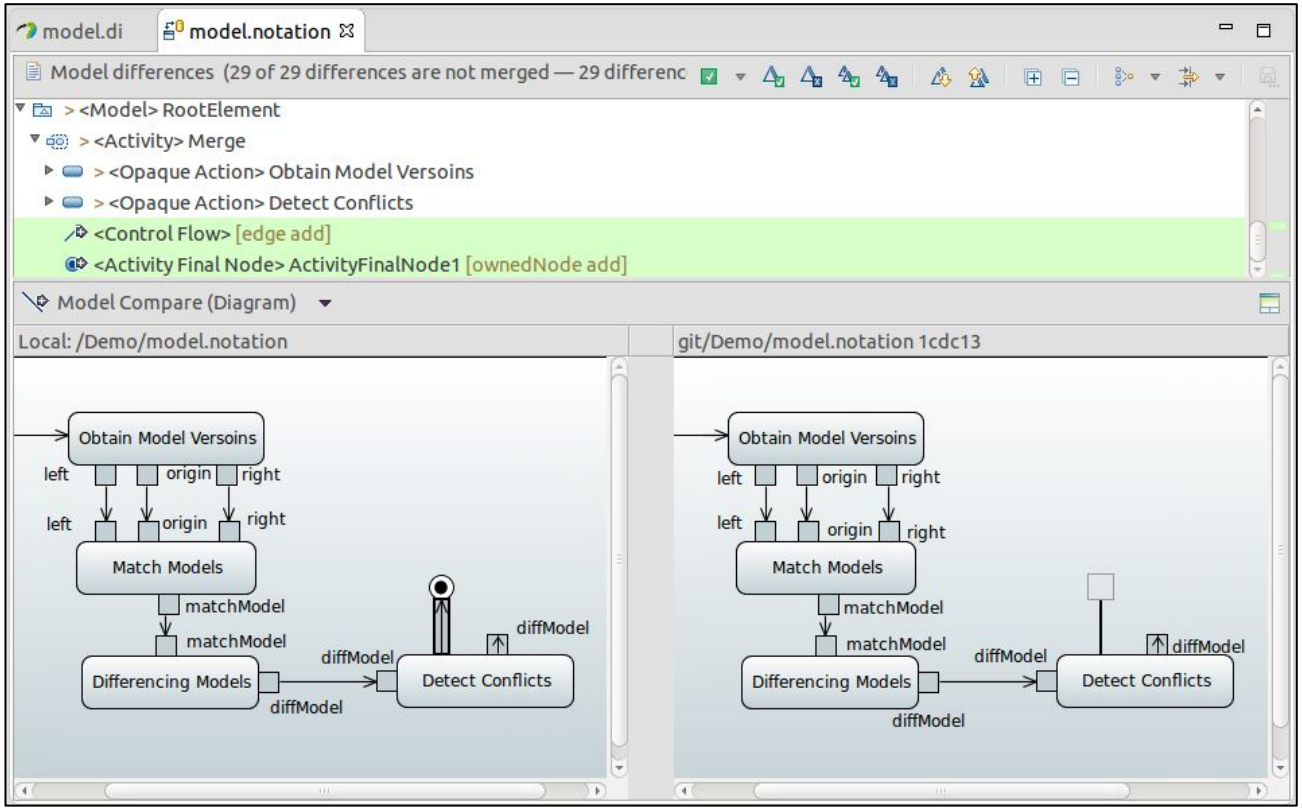


EclipseSource

beo Model Driven Company

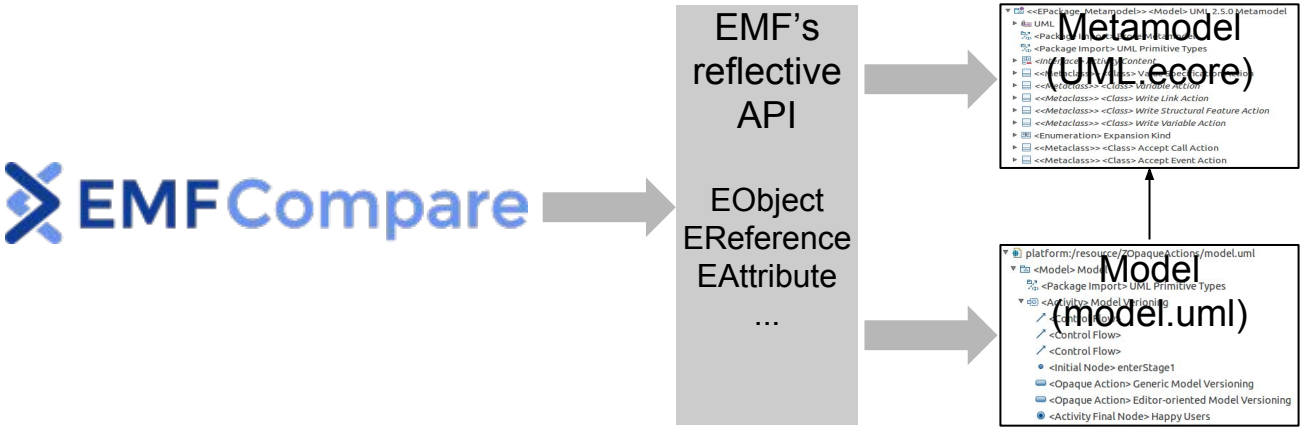
EMF Compare

… should be as close as possible to how you compare/merge them



How does EMF Compare work?

- Raise the unit of comparison from text line to **model elements**
- Compare and merge the **logical structure** of the models
- EMF Compare's core is built with generic algorithms
  - Make use of the metamodel to obtain knowledge on logical structure
  - **Reflective API** to access and compare values generically
  - Support **every modeling language** that is specified in EMF

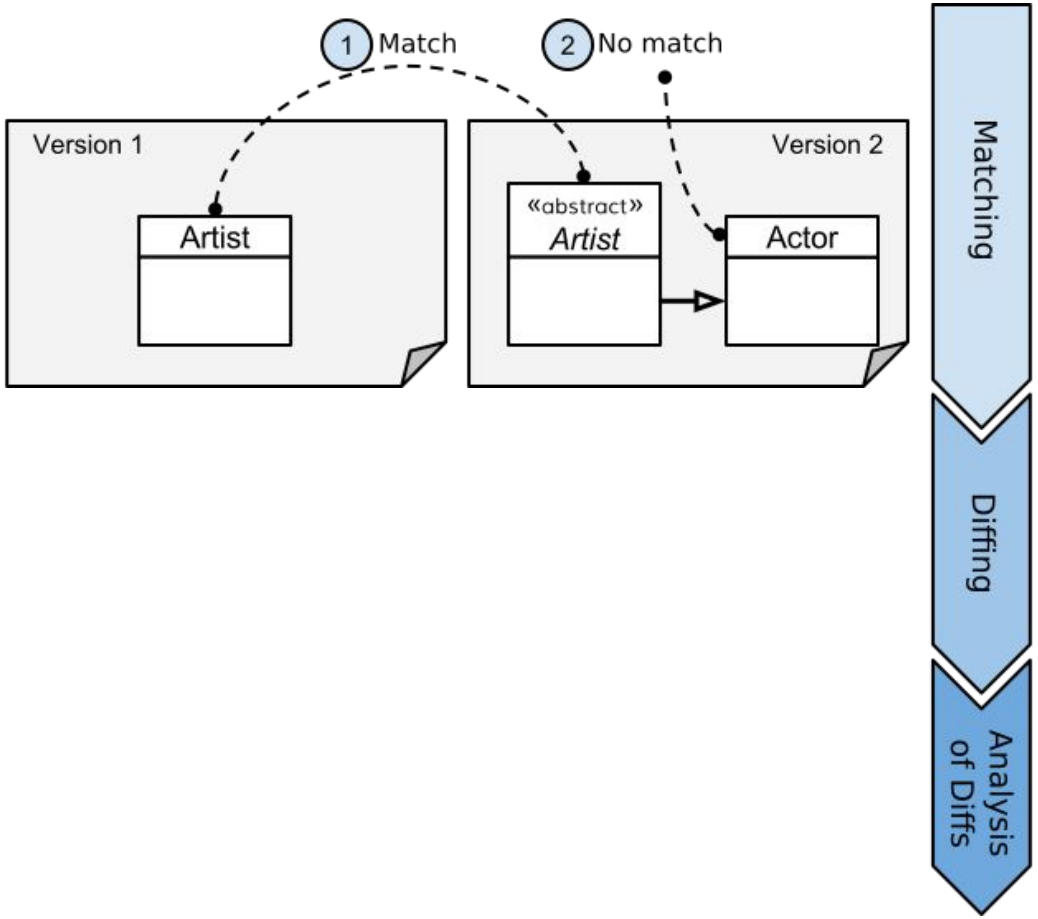


How does EMF Compare work?

- Model comparison phases
  - Matching
    Finding corresponding model elements
  - Diffing
    Identifying differences among corresponding model elements
  - Analysis of differences
    Equivalences, dependencies, and conflicts among differences

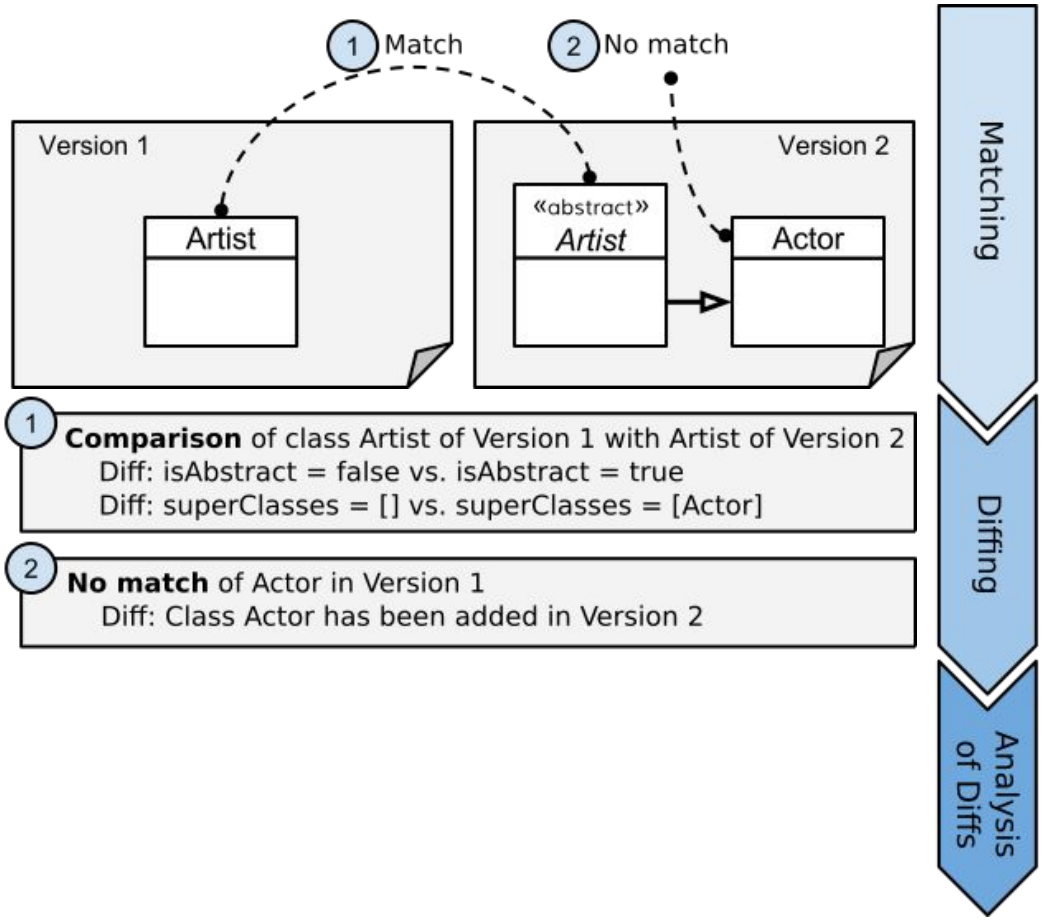
How does EMF Compare work?

- Model comparison phases
  - Matching
    Finding corresponding model elements
  - Diffing
    Identifying differences among corresponding model elements
  - Analysis of differences
    Equivalences, dependencies, and conflicts among differences



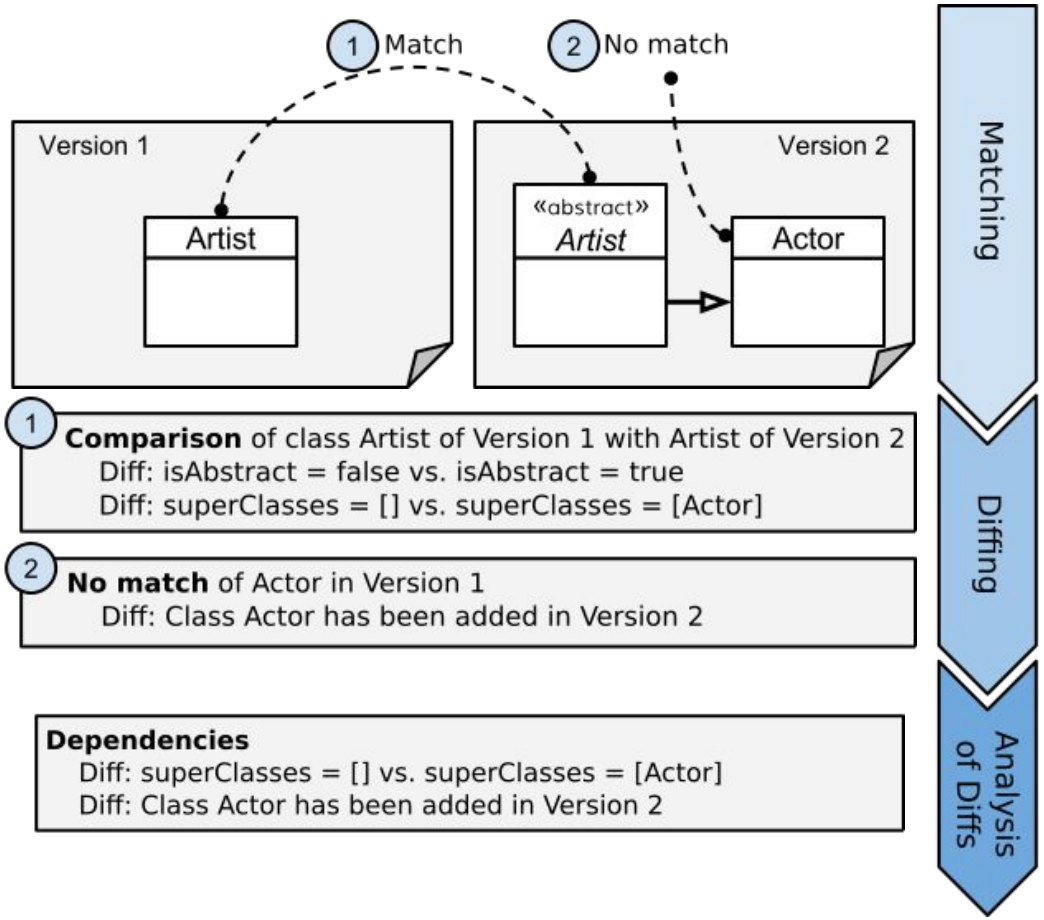
How does EMF Compare work?

- Model comparison phases
  - Matching
    Finding corresponding model elements
  - Diffing
    Identifying differences among corresponding model elements
  - Analysis of differences
    Equivalences, dependencies, and conflicts among differences



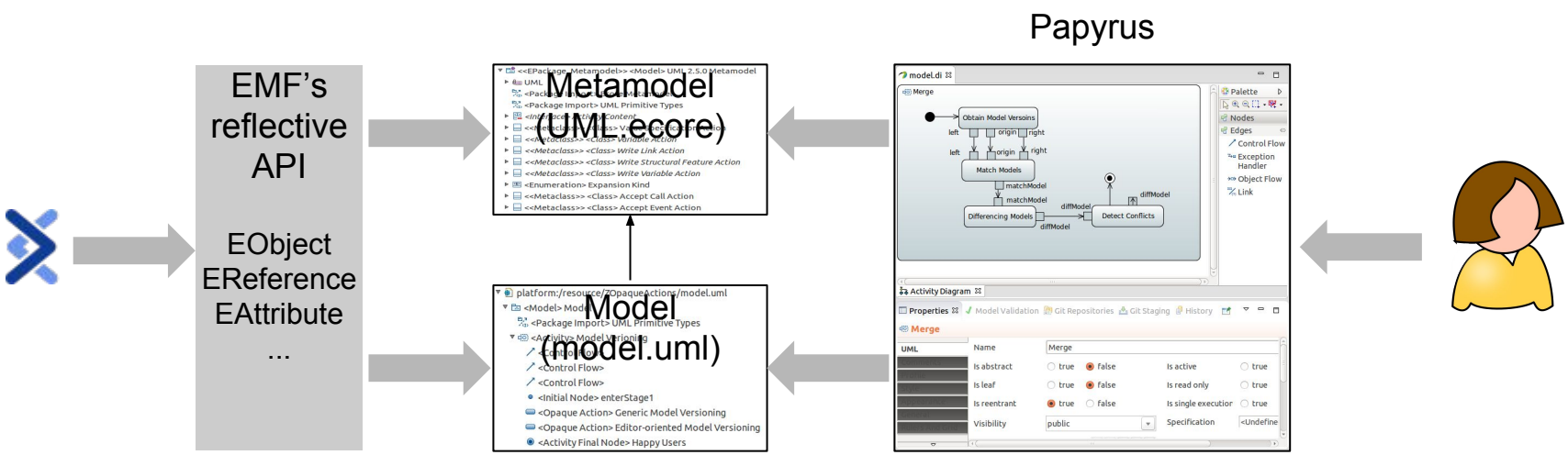
How does EMF Compare work?

- Model comparison phases
  - Matching
    Finding corresponding model elements
  - Diffing
    Identifying differences among corresponding model elements
  - Analysis of differences
    Equivalences, dependencies, and conflicts among differences



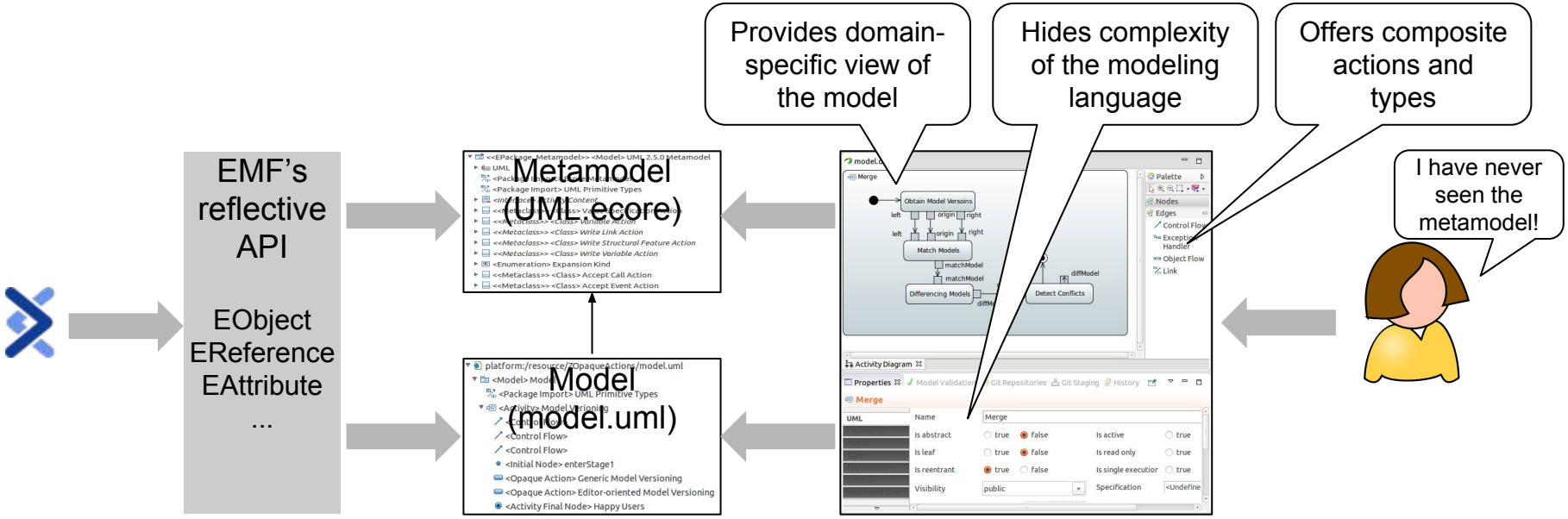
When and why is customization needed?

- Gap between what EMF Compare processes and what users see
  - EMF Compare only “knows” the abstract model and the metamodel
  - Users only know how to use the modeling editor



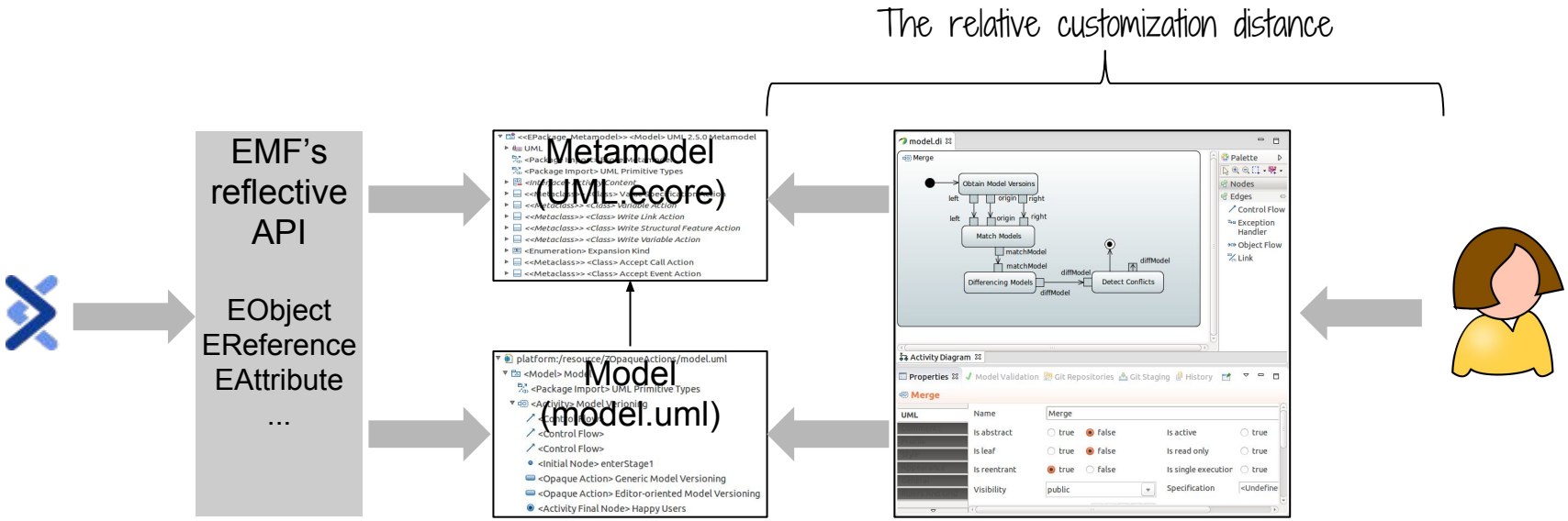
When and why is customization needed?

- Gap between what EMF Compare processes and what users see
  - EMF Compare only “knows” the abstract model and the metamodel
  - Users only know how to use the modeling editor



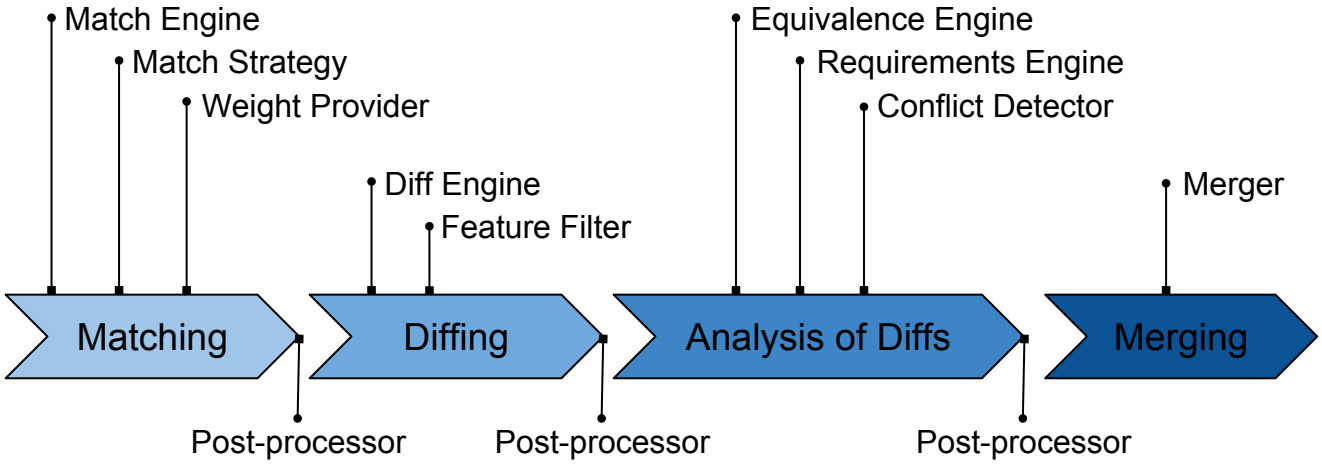
When and why is customization needed?

- Gap between what EMF Compare processes and what users see
  - EMF Compare only “knows” the abstract model and the metamodel
  - Users only know how to use the modeling editor

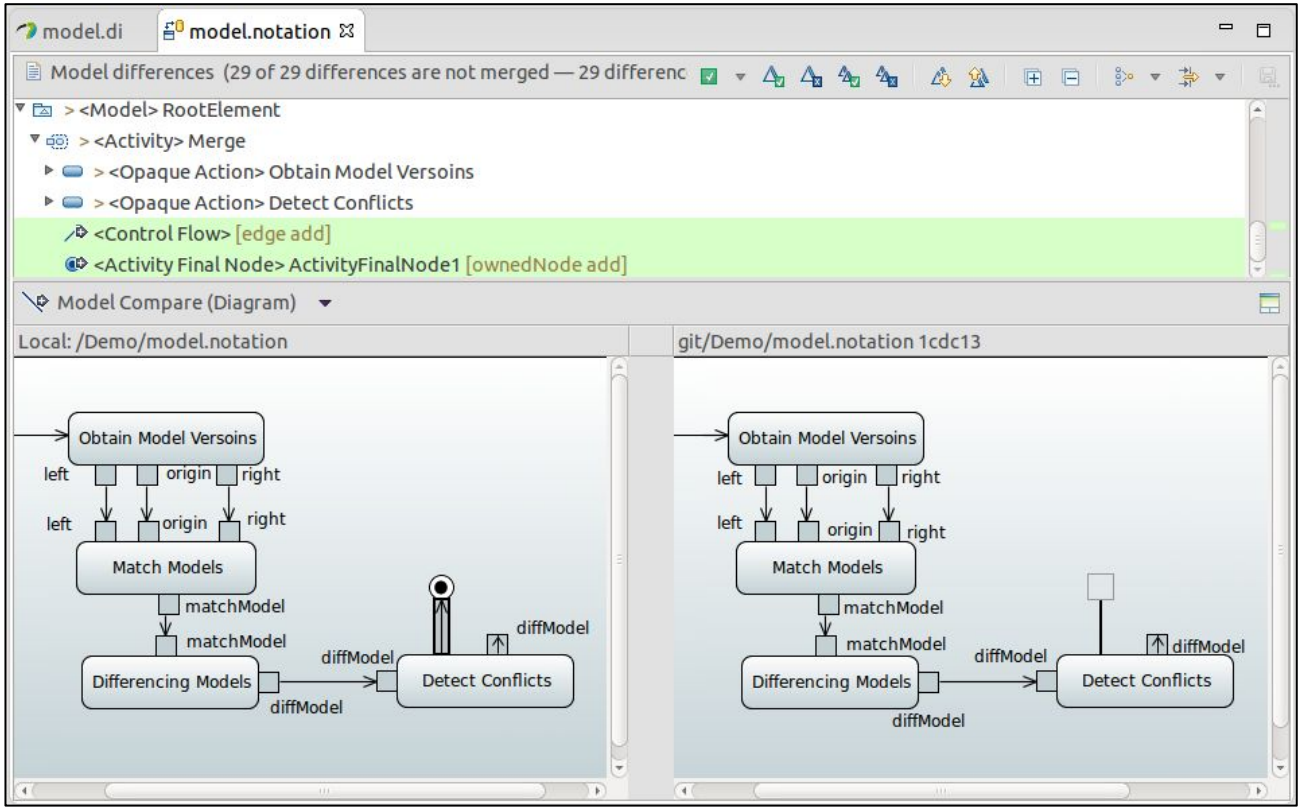


Customizing EMF Compare

- EMF Compare is a highly-customizable framework
  - Enabling us to implement tailor-made model comparison tools
  - Take generic support of EMF Compare and make it domain-specific
- Customization and extension points: EMF Compare Core

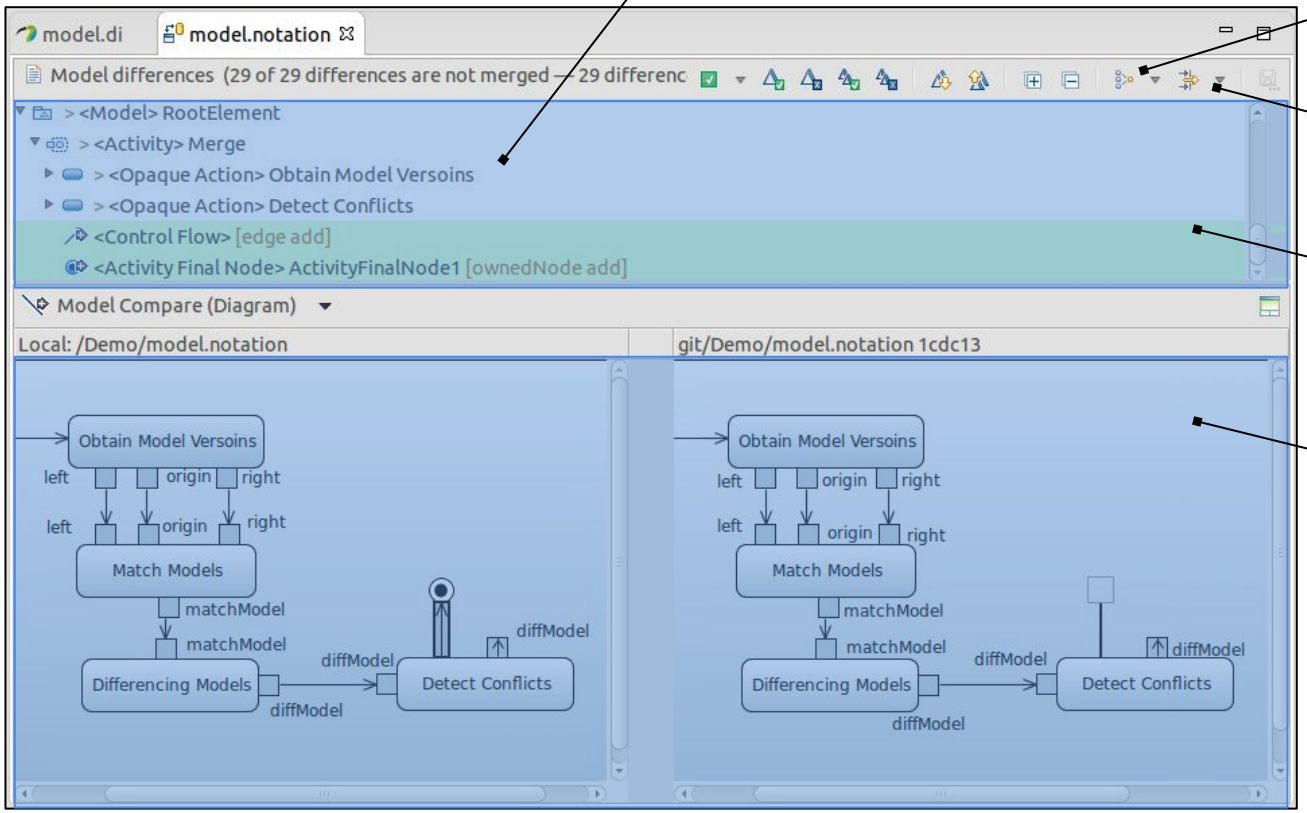


Customizing the EMF Compare Merge Viewer



Customizing the EMF Compare Merge Viewer

Group providers specify the content of this tree over a comparison
Default providers can be customized



Additional group providers
can be added for more options

Filters
can be added and auto-activated or just added as option for users

Elements have to adapt to a type
Double-clicking an element will open the content merge viewer for the respective type

Content merge viewers
One or more content merge viewers can be provided for a defined type

Demos of Example Customizations

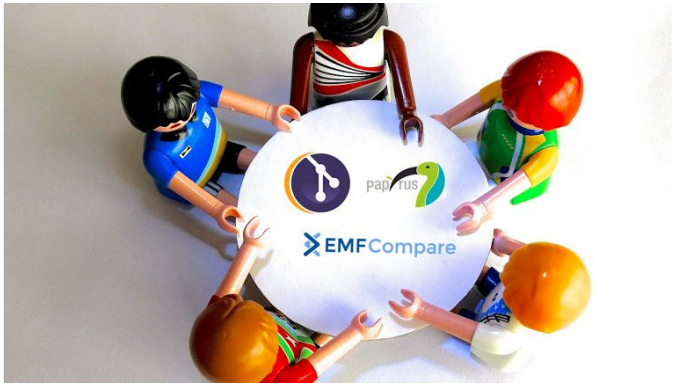
- Library example
  - Additional filter
  - Additional domain-specific grouping option
  - New difference type
- Other model viewer
  - Custom content merge viewer

Future Perspectives

- It isn't possible to build a one-size-fits all model comparison tool
  - Difference viewer should be as close as possible to how users work
  - Customized modeling editor → customized model comparison tool
- Build a highly extensible and well-performing basis
  - Make extending EMF Compare as easy as possible
  - Maximize the reuse of knowledge from the modeling editor
- Build a perfect ready-to-use model comparison tool
  - For Papyrus UML, UML-RT, and SysML on top of EMF Compare
    → http://www.collaborative-modeling.org

Thank you very much!

- More information
  - EMF Compare Developer Guide
  - EMF Compare Forum
  - Contact us
- Follow us on http://www.collaborative-modeling.org



eclipsecon Europe
Ludwigsburg, Germany, 3 - 5 November 2015
Evaluate the sessions at www.eclipsecon.org
+1
0
-1