

EMF Compare ♥ EGit

Seamless Collaborative Modeling

\$ whoami



Laurent Delaigue

 @laurentdelaigue

laurent.delaigue@obeo.fr



Model Driven Company



Philip Langer

 @PhilChip23

planger@eclipsesource.com



Comparing & Merging Code

```
/**
 * Create a MergeOperation object. Initializes the MergeStrategy with the
 * preferred merge strategy, according to preferences.
 *
 * @param repository
 * @param refName
 *         name of a commit which should be merged
 */
public MergeOperation(@NonNull Repository repository,
    @NonNull String refName) {
    this.repository = repository;
    this.refName = refName;
    this.mergeStrategy = Activator.getDefault().getPreferredMergeStrategy();
}

/**
 * Create a MergeOperation object
 *
 * @param repository
 * @param refName
 *         name of a commit which should be merged
 * @param mergeStrategyName
 *         the strategy to use for merge. If not registered, the default
 *         merge strategy according to preferences will be used.
 */
public MergeOperation(@NonNull Repository repository,
    @NonNull String refName,
    @NonNull String mergeStrategyName) {
    this.repository = repository;
    this.refName = refName;
    MergeStrategy strategy = null;
    strategy = MergeStrategy.get(mergeStrategyName);
    this.mergeStrategy = strategy != null ? strategy : Activator.getDefault()
        .getPreferredMergeStrategy();
}
```

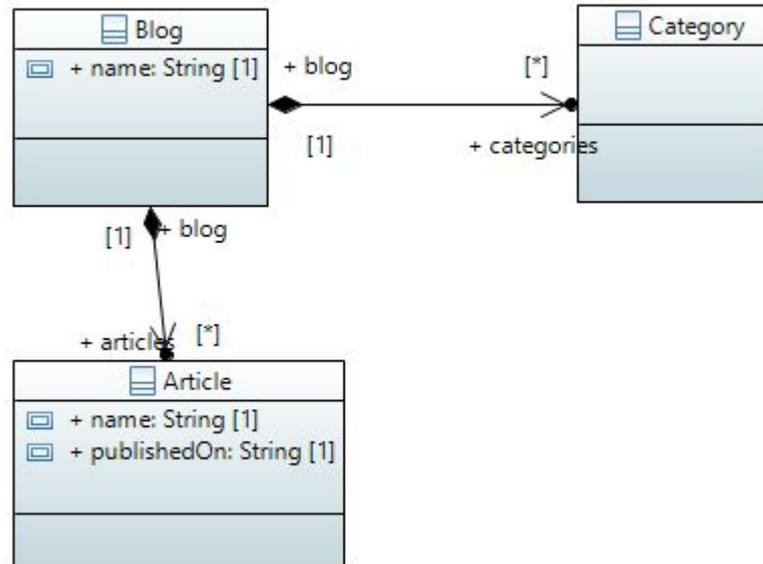
Comparing & Merging Code

```
Java Source Compare
MergeOperation.java 99cdec0... (Laurent Delaigue)
74
75 /**
76  * Create a MergeOperation object. Initializes the MergeStrategy with the
77  * preferred merge strategy, according to preferences.
78  *
79  * @param repository
80  * @param refName
81  *     name of a commit which should be merged
82  */
83 public MergeOperation(@NonNull Repository repository,
84                      @NonNull String refName) {
85     this.repository = repository;
86     this.refName = refName;
87     this.mergeStrategy = Activator.getDefault().getPreferredMergeStrategy();
88 }
89
90 /**
91  * Create a MergeOperation object
92  *
93  * @param repository
94  * @param refName
95  *     name of a commit which should be merged
96  * @param mergeStrategyName
97  *     the strategy to use for merge. If not registered, the default
98  *     merge strategy according to preferences will be used.
99  */
100 public MergeOperation(@NonNull Repository repository,
101                      @NonNull String refName,
102                      @NonNull String mergeStrategyName) {
103     this.repository = repository;
104     this.refName = refName;
105     MergeStrategy strategy = null;
106     strategy = MergeStrategy.get(mergeStrategyName);
107     this.mergeStrategy = strategy != null ? strategy : Activator.getDefault()
108         .getPreferredMergeStrategy();
109 }

MergeOperation.java 3a6a38c... (Thomas Wolf)
71
72 /**
73  * @param repository
74  * @param refName name of a commit which should be merged
75  */
76 public MergeOperation(Repository repository, String refName) {
77     this.repository = repository;
78     this.refName = refName;
79 }
80
81 /**
82  * Create a MergeOperation object
83  * @param repository
84  * @param refName name of a commit which should be merged
85  * @param mergeStrategy the strategy to use for merge
86  */
87 public MergeOperation(Repository repository, String refName,
88                      String mergeStrategy) {
89     this.repository = repository;
90     this.refName = refName;
91     if (mergeStrategy != null)
92         this.mergeStrategy = MergeStrategy.get(mergeStrategy);
93 }
94
95 /**
96  * @param squash true to squash merge commits
97  */
98 public void setSquash(boolean squash) {
99     this.squash = Boolean.valueOf(squash);
100 }
101
102 /**
103  * @param ffmode set the fast forward mode
104  * @since 3.0
105  */
106 public void setFastForwardMode(FastForwardMode ffmode) {
```



Comparing and Merging Models



Comparing and Merging Models

```
Local: model.notation
26 </children>
27 <element xmi:type="uml:Class" href="model.uml#_3yvWwHzCEeW5t_tj7V58QQ"/>
28 <layoutConstraint xmi:type="notation:Bounds" xmi:id="_3zWawXzCEeW5t_tj7V58QQ" x="91" y="76"
29 </children>
30 <children xmi:type="notation:Shape" xmi:id="_4x3RcHzCEeW5t_tj7V58QQ" type="2008">
31 <children xmi:type="notation:DecorationNode" xmi:id="_4x3RcnzCEeW5t_tj7V58QQ" type="5029"/>
32 <children xmi:type="notation:BasicCompartment" xmi:id="_4x3Rc3zCEeW5t_tj7V58QQ" type="7017">
33 <children xmi:type="notation:Shape" xmi:id="_aNxAgHzDEeW5t_tj7V58QQ" type="3012">
34 <element xmi:type="uml:Property" href="model.uml#_aNvYyHzDEeW5t_tj7V58QQ"/>
35 <layoutConstraint xmi:type="notation:Location" xmi:id="_aNxAgXzDEeW5t_tj7V58QQ"/>
36 </children>
37 <children xmi:type="notation:Shape" xmi:id="_c-nXgHzDEeW5t_tj7V58QQ" type="3012">
38 <element xmi:type="uml:Property" href="model.uml#_c-mJYHzDEeW5t_tj7V58QQ"/>
39 <layoutConstraint xmi:type="notation:Location" xmi:id="_c-nXgXzDEeW5t_tj7V58QQ"/>
40 </children>
41 <styles xmi:type="notation:TitleStyle" xmi:id="_4x3RdHzCEeW5t_tj7V58QQ"/>
42 <styles xmi:type="notation:SortingStyle" xmi:id="_4x3RdXzCEeW5t_tj7V58QQ"/>
43 <styles xmi:type="notation:FilteringStyle" xmi:id="_4x3RdnzCEeW5t_tj7V58QQ"/>
44 <layoutConstraint xmi:type="notation:Bounds" xmi:id="_4x3Rd3zCEeW5t_tj7V58QQ"/>
45 </children>
46 <children xmi:type="notation:BasicCompartment" xmi:id="_4x3ReHzCEeW5t_tj7V58QQ" type="7018">
47 <styles xmi:type="notation:TitleStyle" xmi:id="_4x34gHzCEeW5t_tj7V58QQ"/>
48 <styles xmi:type="notation:SortingStyle" xmi:id="_4x34gXzCEeW5t_tj7V58QQ"/>
49 <styles xmi:type="notation:FilteringStyle" xmi:id="_4x34gnzCEeW5t_tj7V58QQ"/>
50 <layoutConstraint xmi:type="notation:Bounds" xmi:id="_4x34g3zCEeW5t_tj7V58QQ"/>
51 </children>
52 <children xmi:type="notation:BasicCompartment" xmi:id="_4x34hHzCEeW5t_tj7V58QQ" visible="f">
53 <styles xmi:type="notation:TitleStyle" xmi:id="_4x34hXzCEeW5t_tj7V58QQ"/>
54 <styles xmi:type="notation:SortingStyle" xmi:id="_4x34hznzCEeW5t_tj7V58QQ"/>
55 <styles xmi:type="notation:FilteringStyle" xmi:id="_4x34h3zCEeW5t_tj7V58QQ"/>
56 <layoutConstraint xmi:type="notation:Bounds" xmi:id="_4x34hHzCEeW5t_tj7V58QQ"/>
57 </children>

Local history: model.notation 27 oct. 2015 16:57:10
26 </children>
27 <element xmi:type="uml:Class" href="model.uml#_3yvWwHzCEeW5t_tj7V58QQ"/>
28 <layoutConstraint xmi:type="notation:Bounds" xmi:id="_3zWawXzCEeW5t_tj7V58QQ" x="91" y="76"
29 </children>
30 <children xmi:type="notation:Shape" xmi:id="_4x3RcHzCEeW5t_tj7V58QQ" type="2008">
31 <children xmi:type="notation:DecorationNode" xmi:id="_4x3RcnzCEeW5t_tj7V58QQ" type="5029">
32 <children xmi:type="notation:BasicCompartment" xmi:id="_4x3Rc3zCEeW5t_tj7V58QQ" type="7017">
33 <styles xmi:type="notation:TitleStyle" xmi:id="_4x3RdHzCEeW5t_tj7V58QQ"/>
34 <styles xmi:type="notation:SortingStyle" xmi:id="_4x3RdXzCEeW5t_tj7V58QQ"/>
35 <styles xmi:type="notation:FilteringStyle" xmi:id="_4x3RdnzCEeW5t_tj7V58QQ"/>
36 <layoutConstraint xmi:type="notation:Bounds" xmi:id="_4x3Rd3zCEeW5t_tj7V58QQ"/>
37 </children>
38 <children xmi:type="notation:BasicCompartment" xmi:id="_4x3ReHzCEeW5t_tj7V58QQ" type="7018">
39 <styles xmi:type="notation:TitleStyle" xmi:id="_4x34gHzCEeW5t_tj7V58QQ"/>
40 <styles xmi:type="notation:SortingStyle" xmi:id="_4x34gXzCEeW5t_tj7V58QQ"/>
41 <styles xmi:type="notation:FilteringStyle" xmi:id="_4x34gnzCEeW5t_tj7V58QQ"/>
42 <layoutConstraint xmi:type="notation:Bounds" xmi:id="_4x34g3zCEeW5t_tj7V58QQ"/>
43 </children>
44 <children xmi:type="notation:BasicCompartment" xmi:id="_4x34hHzCEeW5t_tj7V58QQ" visible="f">
45 <styles xmi:type="notation:TitleStyle" xmi:id="_4x34hXzCEeW5t_tj7V58QQ"/>
46 <styles xmi:type="notation:SortingStyle" xmi:id="_4x34hznzCEeW5t_tj7V58QQ"/>
47 <styles xmi:type="notation:FilteringStyle" xmi:id="_4x34h3zCEeW5t_tj7V58QQ"/>
48 <layoutConstraint xmi:type="notation:Bounds" xmi:id="_4x34hHzCEeW5t_tj7V58QQ"/>
49 </children>
50 <element xmi:type="uml:Class" href="model.uml#_4x2DUHzCEeW5t_tj7V58QQ"/>
51 <layoutConstraint xmi:type="notation:Bounds" xmi:id="_4x3RcXzCEeW5t_tj7V58QQ" x="91" y="76"
52 </children>
53 <children xmi:type="notation:Shape" xmi:id="_NLPLyHzDEeW5t_tj7V58QQ" type="2008">
54 <children xmi:type="notation:DecorationNode" xmi:id="_NLQMcHzDEeW5t_tj7V58QQ" type="5029">
55 <children xmi:type="notation:BasicCompartment" xmi:id="_NLQMcXzDEeW5t_tj7V58QQ" type="7017">
56 <styles xmi:type="notation:TitleStyle" xmi:id="_NLQMcHzDEeW5t_tj7V58QQ"/>
57 <styles xmi:type="notation:SortingStyle" xmi:id="_NLQMc3zDEeW5t_tj7V58QQ"/>
```



Let's bring git to graphical modelers!

Goal

Allow teams to
work on
graphical models
in a
collaborative manner

Context



open-source



***Generic Editor
for Graphical Syntax***



***Full-featured UML
modeler in eclipse***



Compare and merge EMF models



JGit: Java Implementation of Git



EGit: Integration of JGit in eclipse

XML/XMI and Textual Merge

After all, XML files are textual files...

Why should there be a problem to merge them?

XML/XMI & Textual Merge - Human Factors

Users are not UML or EMF experts

Metamodel???

Hundreds of concepts in UML

Users don't want to see XML/XMI

That's NOT what they manipulate in the editor, why should they see that when merging?

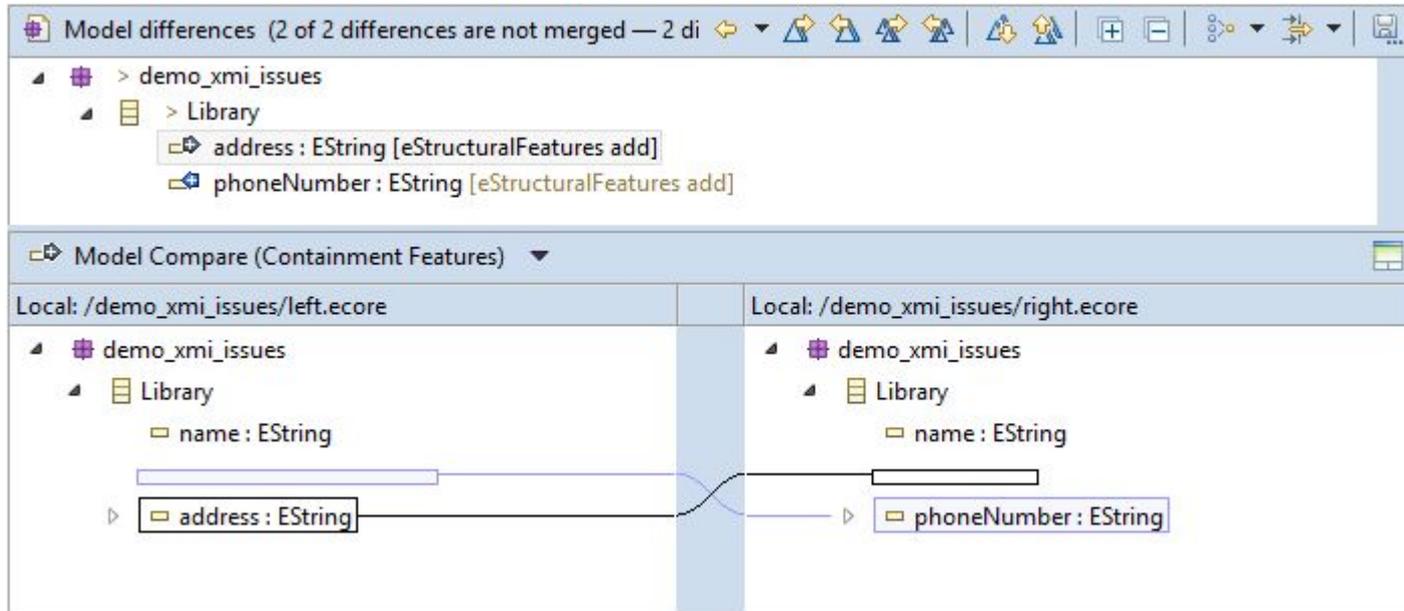
To many users, XMI is as clear as binary

**Users cannot be aware
of the tool's logic and internal rules**

Tools hide complexity

Separation of semantics and representation

XML/XMI & Textual Merge - Technical Factors



```
<eClassifiers xsi:type="ecore:EClass" >
  <eStructuralFeatures xsi:type="ecore:EString" >
    <eStructuralFeatures xsi:type="ecore:EString" >
      </eClassifiers>
</ecore:EPackage>
```

```
<eClassifiers xsi:type="ecore:EClass" >
  <eStructuralFeatures xsi:type="ecore:EString" >
    <eStructuralFeatures xsi:type="ecore:EString" >
      </eClassifiers>
</ecore:EPackage>
```

XML/XMI & Textual Merge - Technical Factors

The screenshot displays a software interface for comparing two models. At the top, a status bar indicates "Model differences (27 of 27 differences are not merged — 0 differences filtered from view)". Below this, a tree view shows the structure of the models being compared.

Model Differences:

- platform:/resource/Test_ECE15/Blog.uml <->
 - <Package> Blog [controlled in platform:/resource/Test_ECE15/Blog.uml]

Model Compare (Containment Features):

Local: /Test_ECE15/Blog.uml	Local: /Test_ECE15/model.uml
<ul style="list-style-type: none">platform:/resource/Test_ECE15/Blog.uml<ul style="list-style-type: none"><Package> Blog<ul style="list-style-type: none"><Class> Category<Class> Article<Association> A_blog_categories<Class> Blog<Association> A_blog_articles	<ul style="list-style-type: none">platform:/resource/Test_ECE15/model.uml<ul style="list-style-type: none"><Package> Blog<ul style="list-style-type: none"><Class> Category<Class> Article<Association> A_blog_categories<Class> Blog<Association> A_blog_articles

Test_ECE15/Copy of model.uml

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <uml:Model xmi:version="20131001" xmlns:xmi="http://www.omg.org/spec/XMI/20131001" xmi:id="1"
3   <packagedElement xmi:type="uml:Package" href="Blog."
4 </uml:Model>
5
```

Test_ECE15/model.uml

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <uml:Model xmi:version="20131001" xmlns:xmi="http://www.omg.org/spec/XMI/20131001" xmi:id="1"
3   <packagedElement xmi:type="uml:Package" xmi:id="1" href="Blog."
4     <packagedElement xmi:type="uml:Class" xmi:id="2" href="Category."
5     <packagedElement xmi:type="uml:Class" xmi:id="3" href="Article."
6       <ownedAttribute xmi:type="uml:Property" xmi:id="1" href="A_blog_categories."
7         <type xmi:type="uml:PrimitiveType" href="http://www.omg.org/spec/XMI/20131001/PrimitiveTypes.xmi#String"
8       </ownedAttribute>
9     <ownedAttribute xmi:type="uml:Property" xmi:id="2" href="A_blog_articles."
10      <type xmi:type="uml:PrimitiveType" href="http://www.omg.org/spec/XMI/20131001/PrimitiveTypes.xmi#String"

```

Logical Model

Ensuring Consistency

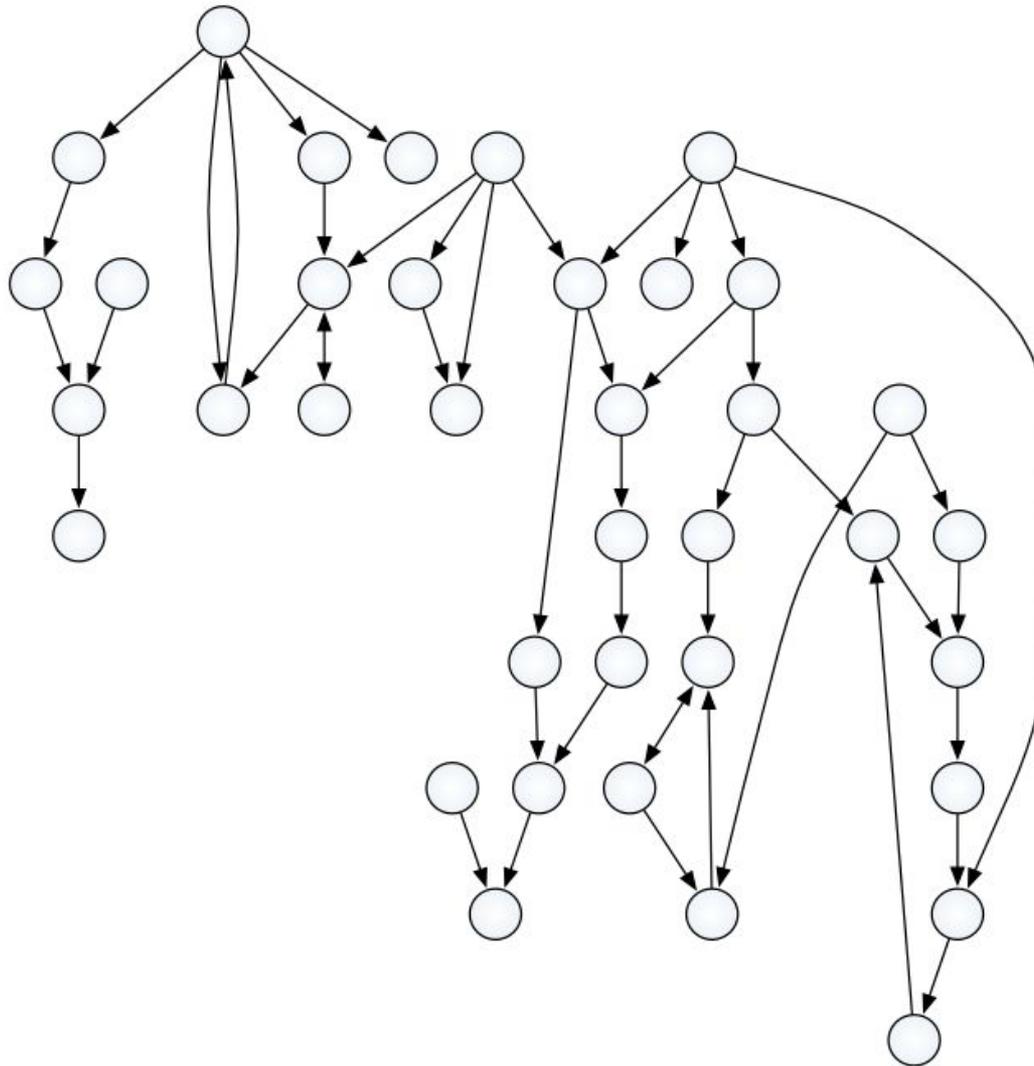
What?

Materialize the fact that a model is made of several (numerous) files

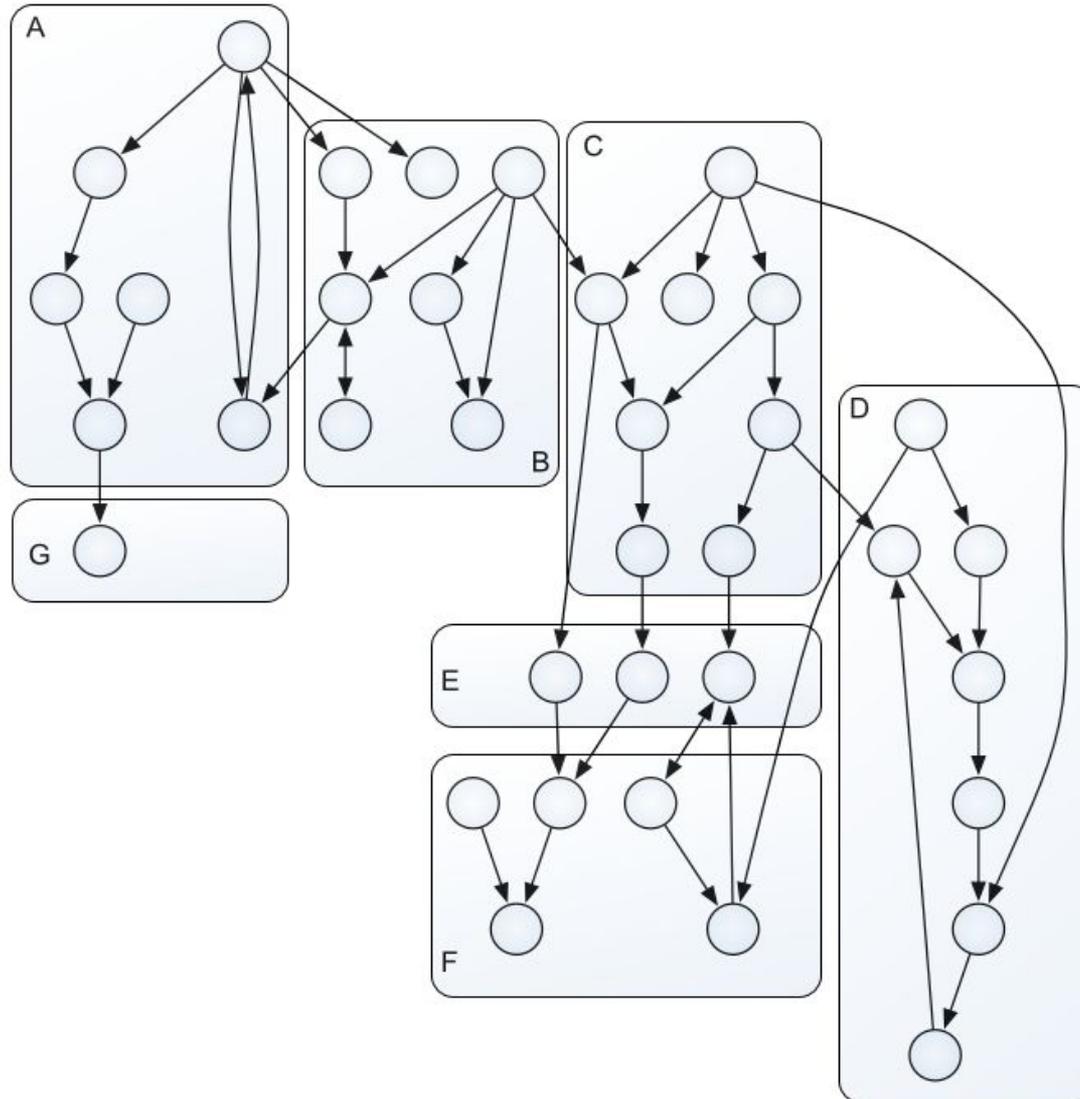
Why?

Keep consistency of models during commits, merges, ...

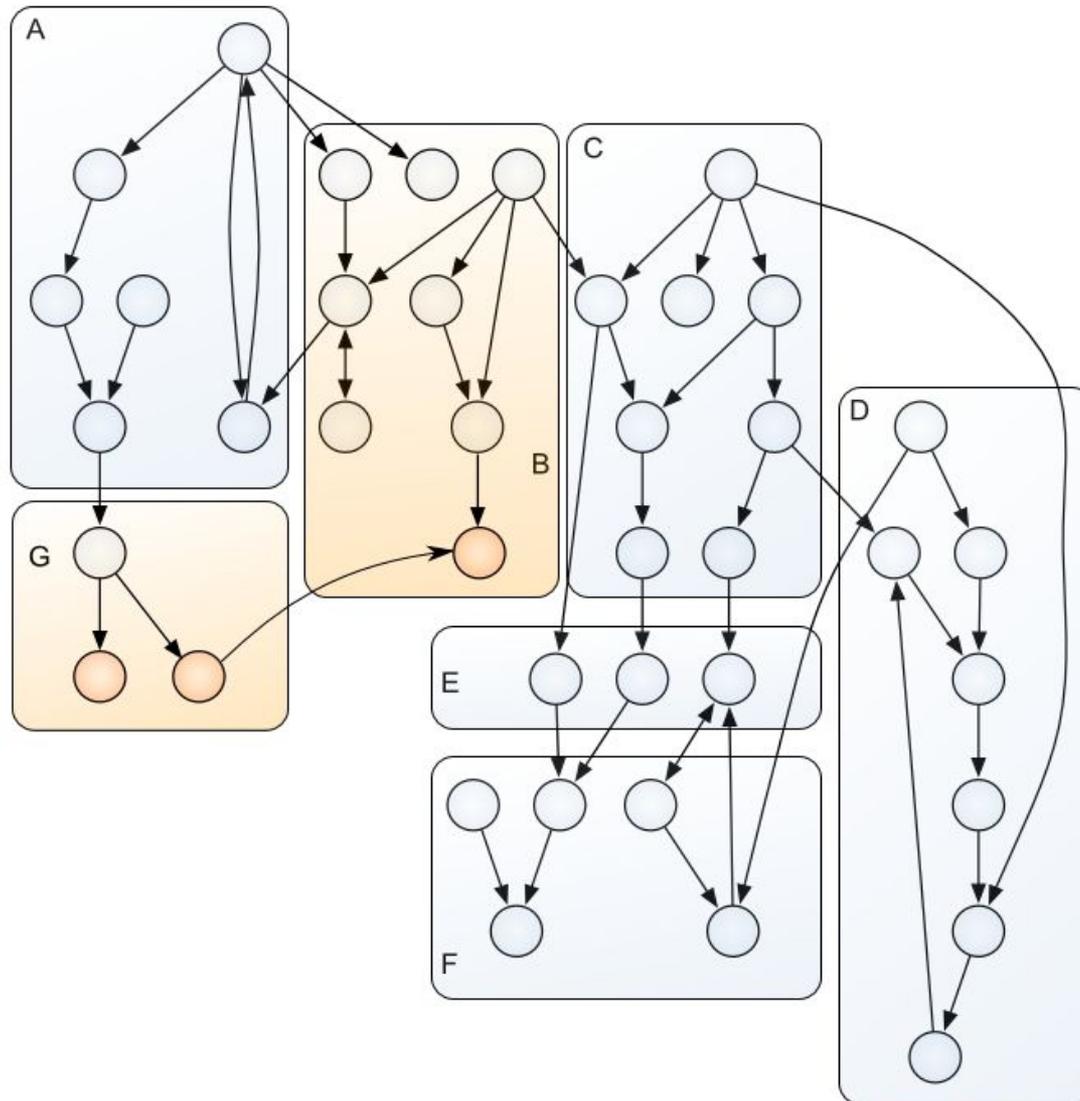
Logical Model - Definition



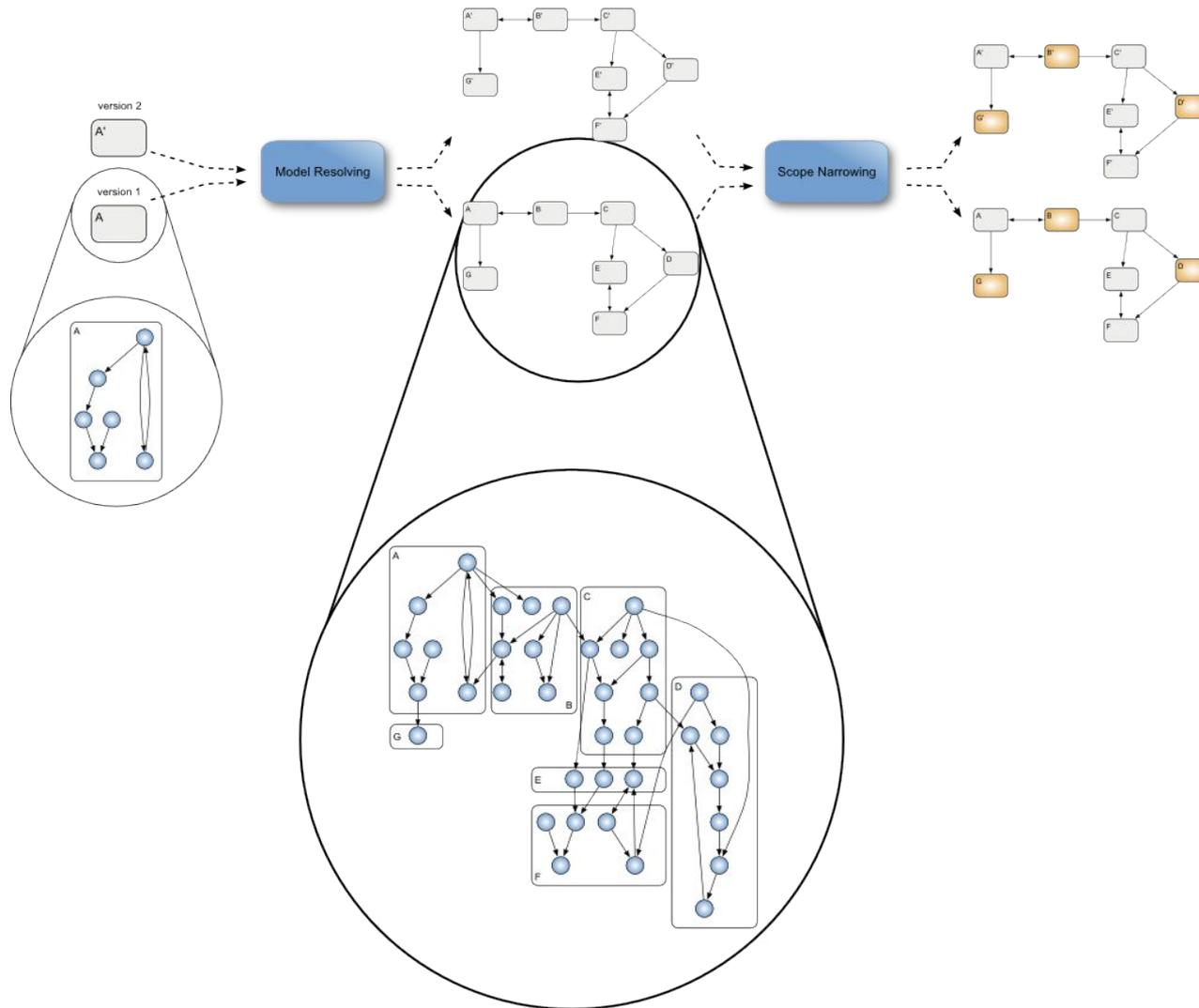
Logical Model - Definition



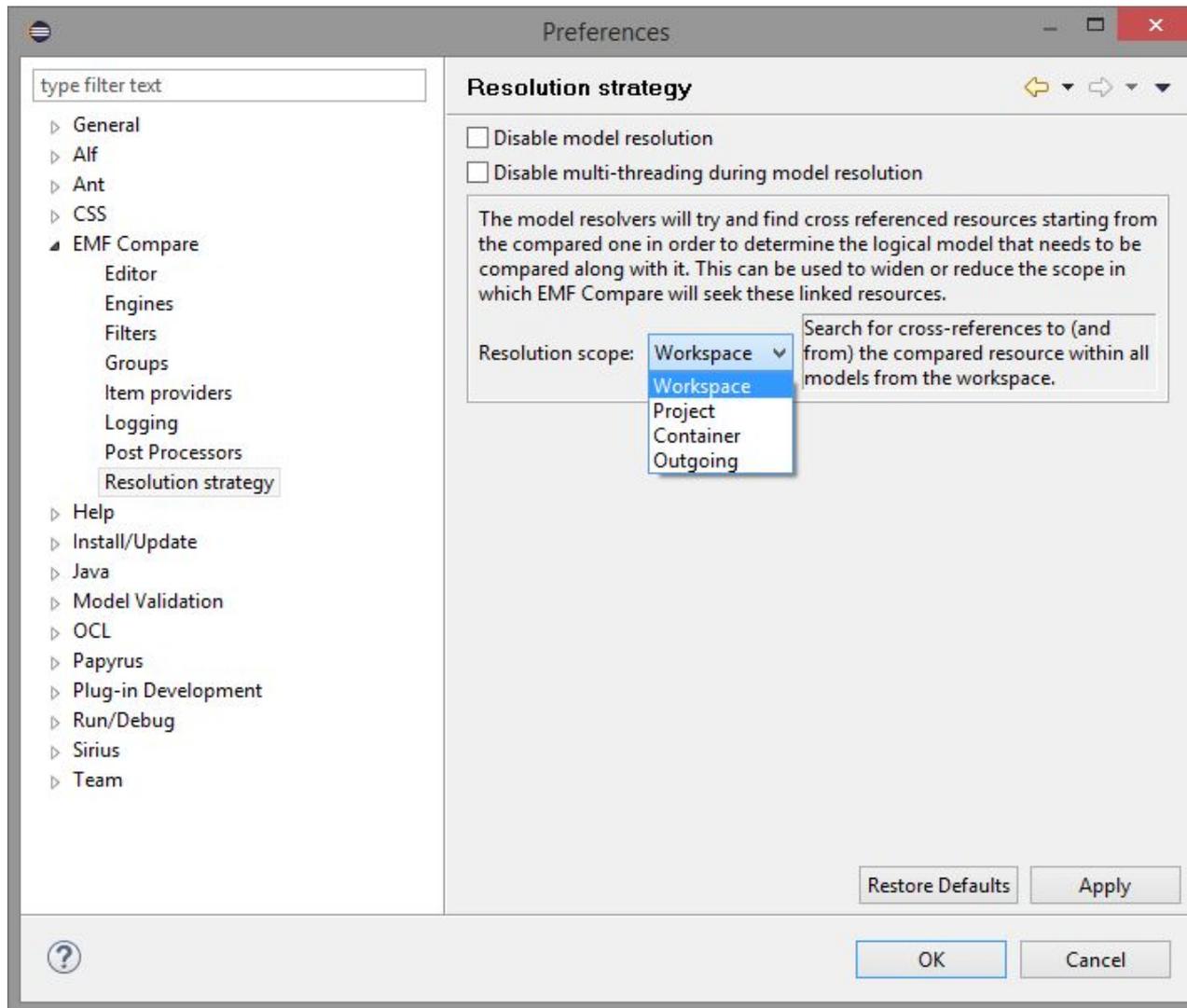
Logical Model - Application



Logical Model - Application



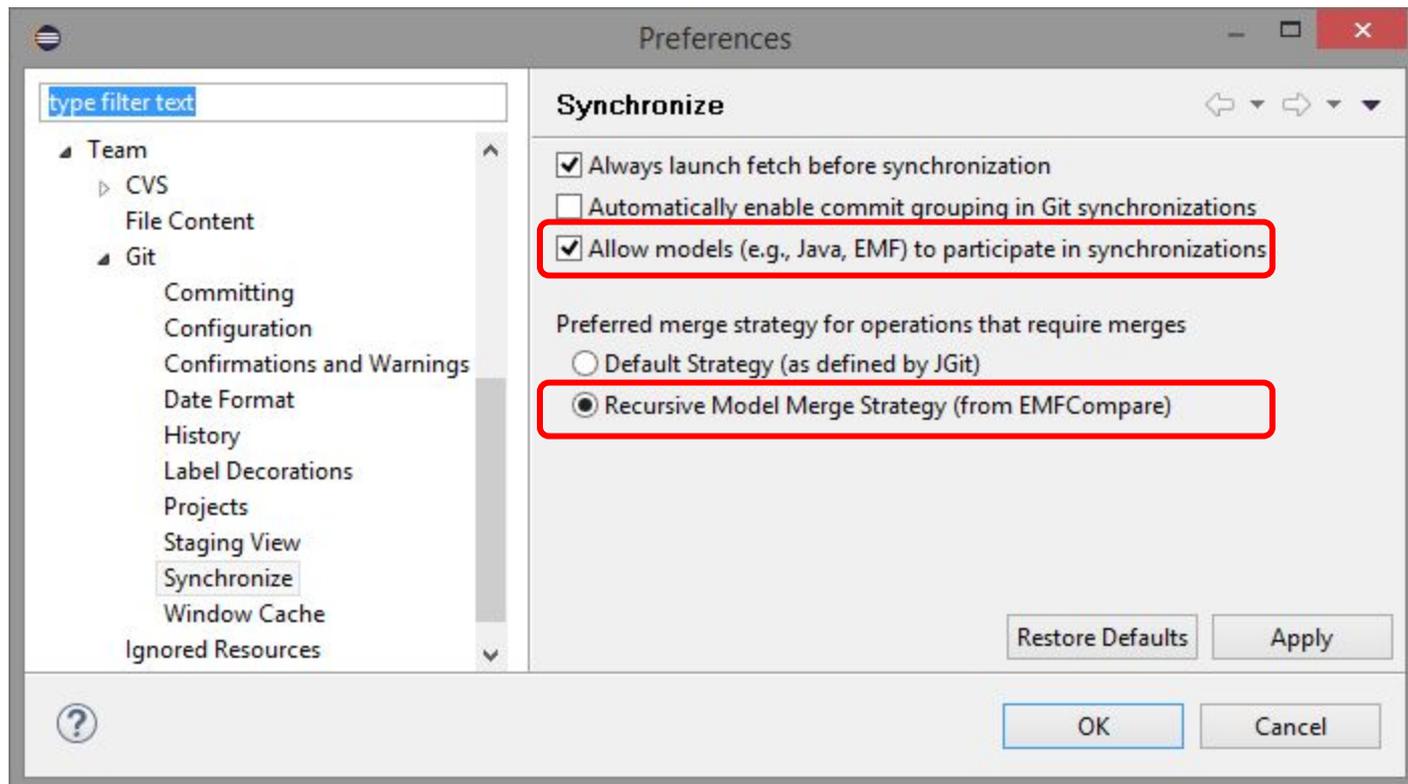
Logical Model - Scope Configuration



Logical Model - Git Interaction

Need to **compare/merge** all the files together

⇒ Compute the logical model for each version (Local, remote, common ancestor)



User Stories

* SEARCH
AS A USER I WOULD
LIKE TO BE ABLE TO
FIND STUFF BY NAME
LOCATION, ETC, ETC
SM

* COMMENT
AS A USER I WOULD
LIKE TO BE ABLE TO
DISCUSS / ASK QUESTIONS
AGAINST A PAGE SO I
DON'T HAVE TO CHANGE IT
TO CONTRIBUTE
MM

* SPAM
AS A USER I WOULD
LIKE TO BE ABLE TO
REPORT SPAM
NEXT LE

* ANSWER / IDENTITY
AS A USER I WOULD LIKE TO
BE ABLE TO MANAGE
WHERE IDENTITIES AND
HOW THEY ARE
COMPANY, PAYING CUSTOMER,
RANDOM USER ETC

* EDITING
AS A USER I WOULD LIKE
TO BE ABLE TO EASILY
EDIT A PAGE SO I
CAN KEEP IT UP TO
DATE

* TAGGING
AS A USER I WOULD
LIKE TO BE ABLE TO
TAG A PAGE
FIND IT
IT

* CREATING
AS A USER I WOULD LIKE
CONTRIBUTORS TO BE
CLEARLY LISTED SO
I CAN
AND MANAGE THEM
TO LIST

* AUDIT TRAIL
AS A USER I WOULD
LIKE TO SEE THE HISTORY
OF CHANGES
EDITED

* ADDING A COMMENT
AS A USER I WOULD
LIKE TO BE ABLE TO ADD A
COMMENT TO THE PAGE
(AND BE GIVEN AN ID)
SO I CAN LINK TO IT

* FEEDS
AS A USER / CUSTOMER
I WOULD LIKE A RSS
FEED FOR CHANGES
TO A PAGE / FEEDBACK
THE WHOLE SITE

* PROFILE PAGE
AS A USER I
WOULD LIKE A
PROFILE PAGE
WITH RSS FEEDS
ETC

* PROFILE PAGE
AS A USER I WOULD LIKE
TO COMMUNICATE
WITH OTHERS
WHICH I HAVE
RECOMMENDED BY OTHERS
(AND) (AND)
EASILY
SEE MY
DATA

* DATA EXPORT
AS A USER I WOULD
LIKE TO BATCH
EXPORT COMMENTS
ETC
SO I CAN DELETE
LARGE AMOUNTS OF DATA

* CLAIM
AS THE OWNER OF A
COMMENT
I WOULD LIKE TO CLAIM
MY COMMENT PAGE
TO BE AN EDITORIAL
EDITOR

Story 1

Logical Model and committing

As a user,

I want to commit all my related changes at a time

So that I don't corrupt my model in any commit

Story 2

Textual Merge
vs
Model Merge

Simple Cases

As a user,

I want to apply my local modifications on top of the master branch

So that I can then benefit from master changes

Story 3

Textual Merge
vs
Model Merge

Complex Cases

As a user

I want to compare my local work (extracted a package in a new project and changed the content) with the master branch

So that I know if my work can be merged on master

Story 4

Conflict management

As a user,

I want to resolve conflicts between my work and the master branch

So that I can merge my work on master



Perspectives

Deploy and collect feedback

Better integration with EGit

Specific build require for now...

Improve Performance

Improve Usability

Make Tool support more generic

Profiles, Other graphical notations



Thank you!



eclipsecon Europe

Ludwigsburg, Germany, 3 - 5 November 2015

Evaluate the sessions at www.eclipsecon.org

+1

0

-1

Useful links

HVOLL

<http://www.eclipse.org/egit/>

<https://www.eclipse.org/emf/compare/>

<http://www.collaborative-modeling.org/>

<https://www.eclipse.org/papyrus/>

<http://www.eclipse.org/sirius/>



Credits

Mona Lisa - https://fr.wikipedia.org/wiki/La_Joconde

by Leonardo da Vinci

Public Domain

The Scream - https://fr.wikipedia.org/wiki/Le_Cri

by Edvard Munch

Public Domain

User Stories - <https://www.flickr.com/photos/psd/3731275681/>

[cc by 2.0](https://creativecommons.org/licenses/by/2.0/)

<https://creativecommons.org/licenses/by/2.0/>

Perspectives <https://unsplash.com/photos/T72-kQyQxtA>

by Jeffrey Swanson

[cc0 1.0](https://creativecommons.org/licenses/by/2.0/) (Public Domain)

<http://creativecommons.org/publicdomain/zero/1.0/>

One last thg - <https://unsplash.com/photos/RbbdzZBKRDY>

by Nitish Meena

[cc0 1.0](https://creativecommons.org/licenses/by/2.0/) (Public Domain)

Useful links - <https://unsplash.com/photos/FDkT3DpYBol>

by Topich

[cc0 1.0](https://creativecommons.org/licenses/by/2.0/) (Public Domain)