

Edapt / COPE

Framework for Instance Migration in Response to Ecore Model Adaptation

Markus Herrmannsdörfer

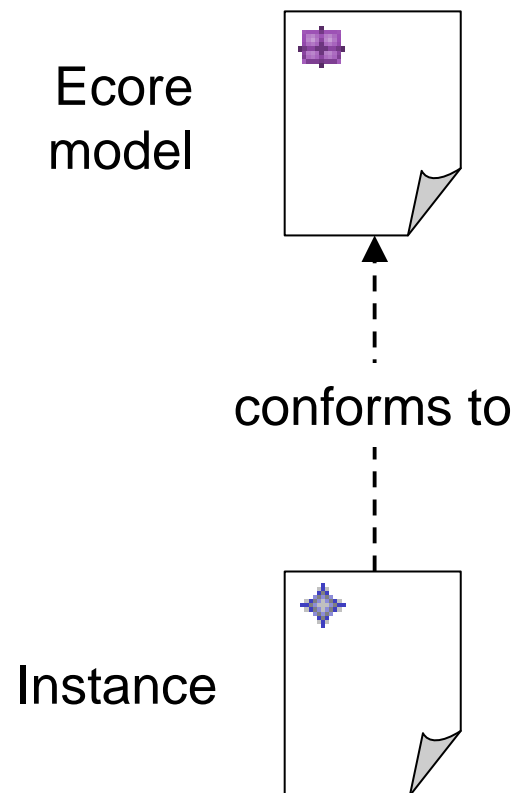
Ignite Talk

EclipseCon 2011

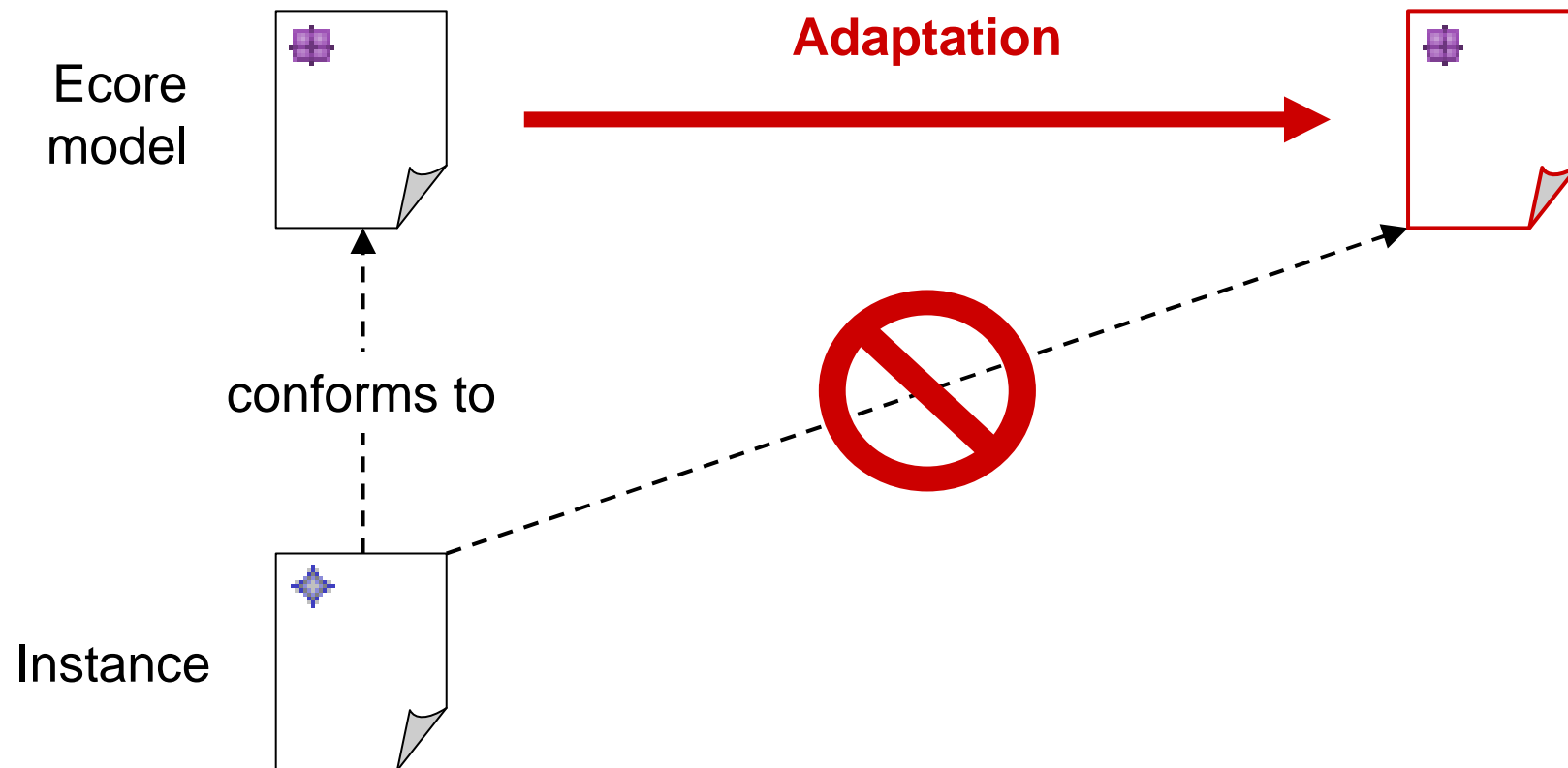
San Francisco, USA

21st March 2011

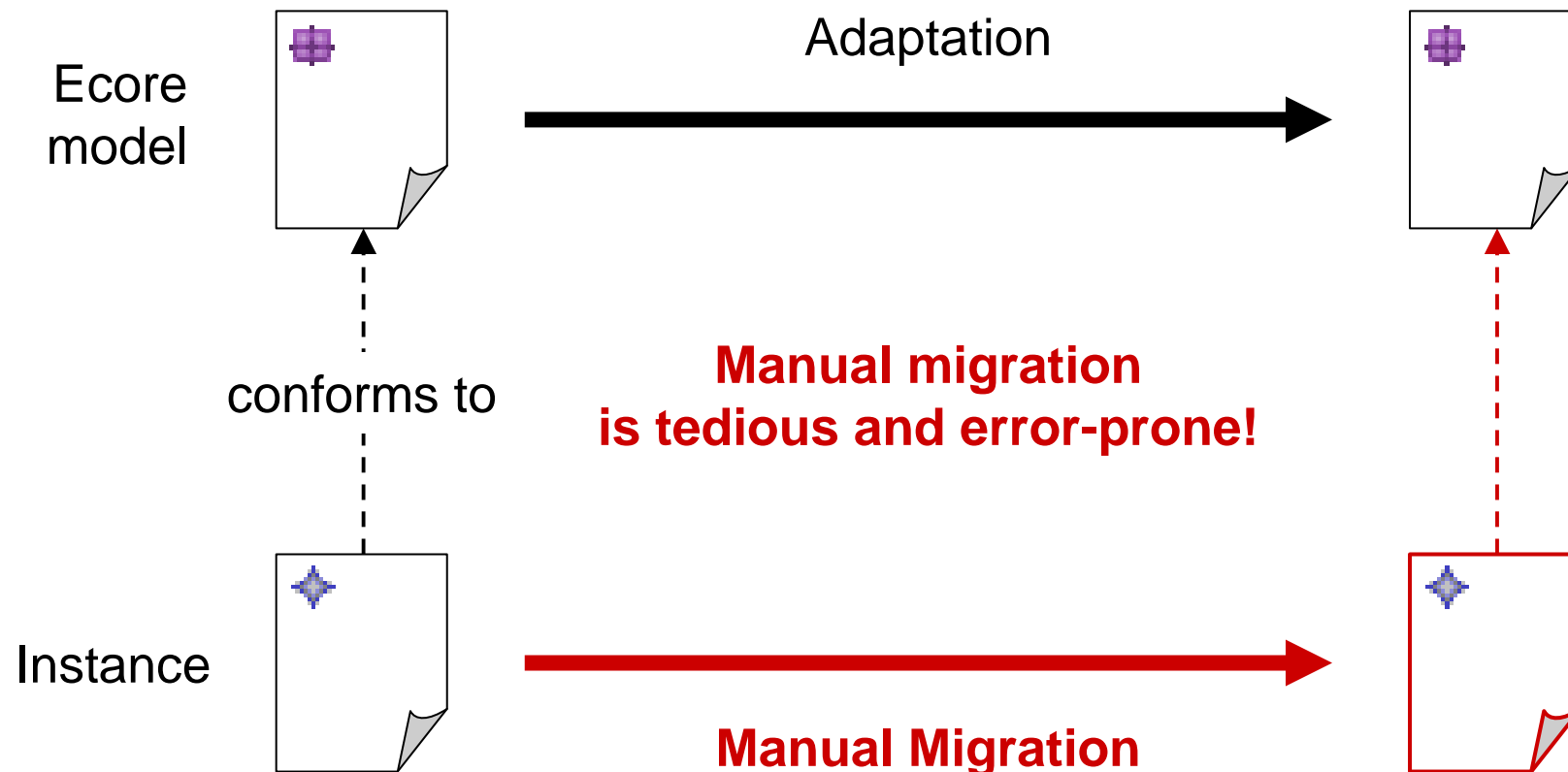
Conformance



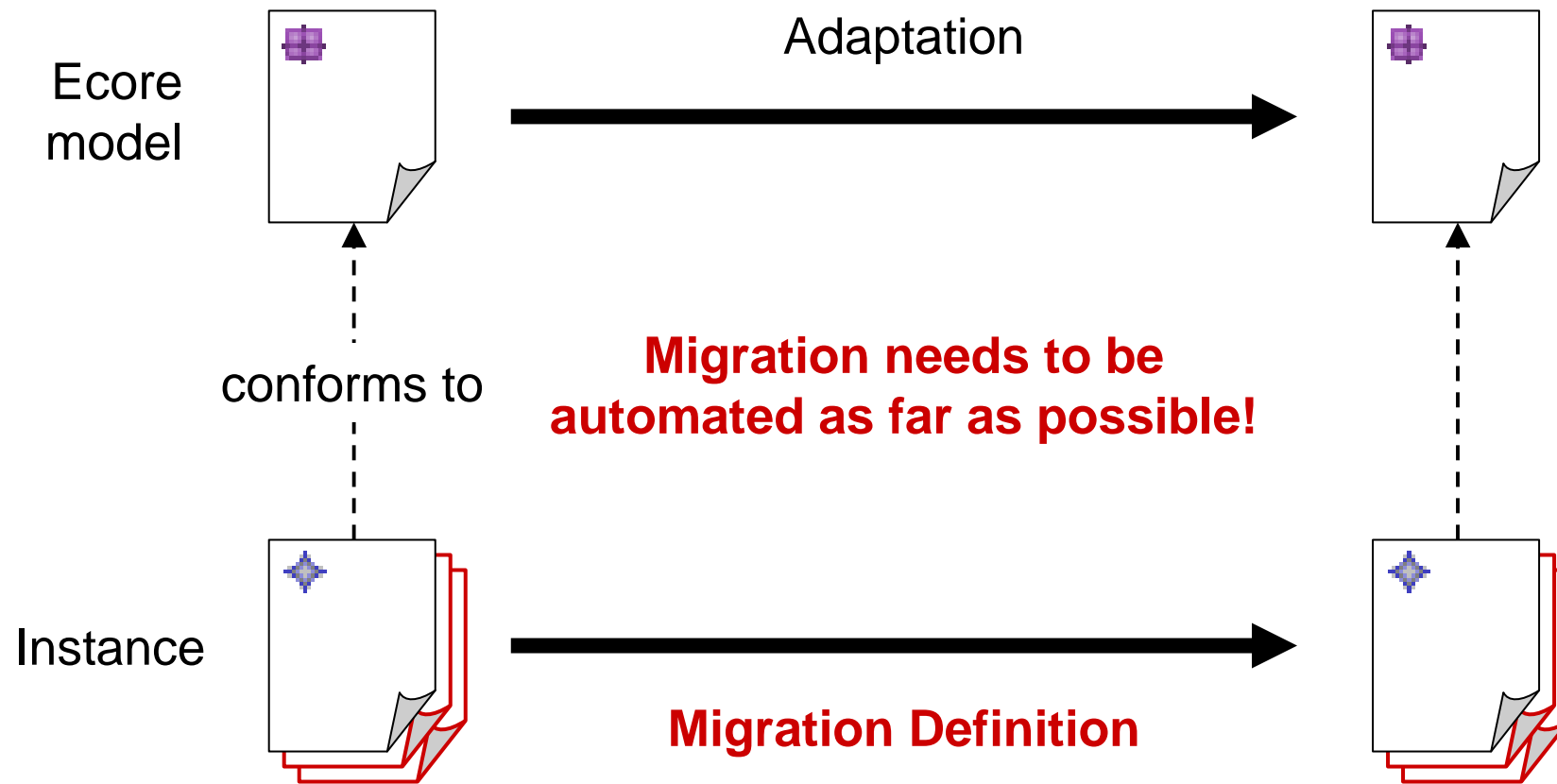
Ecore Model Adaptation



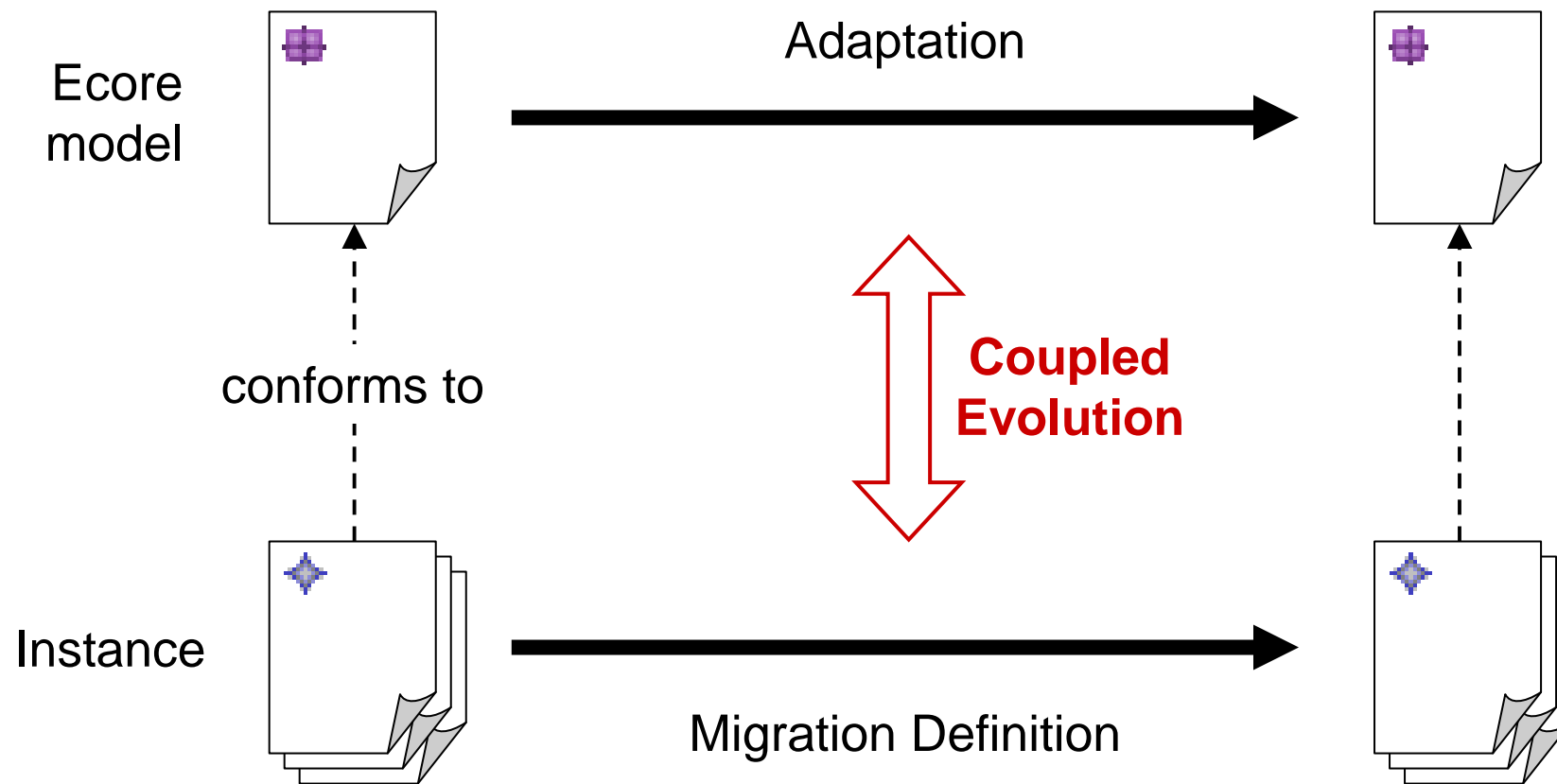
Instance Migration



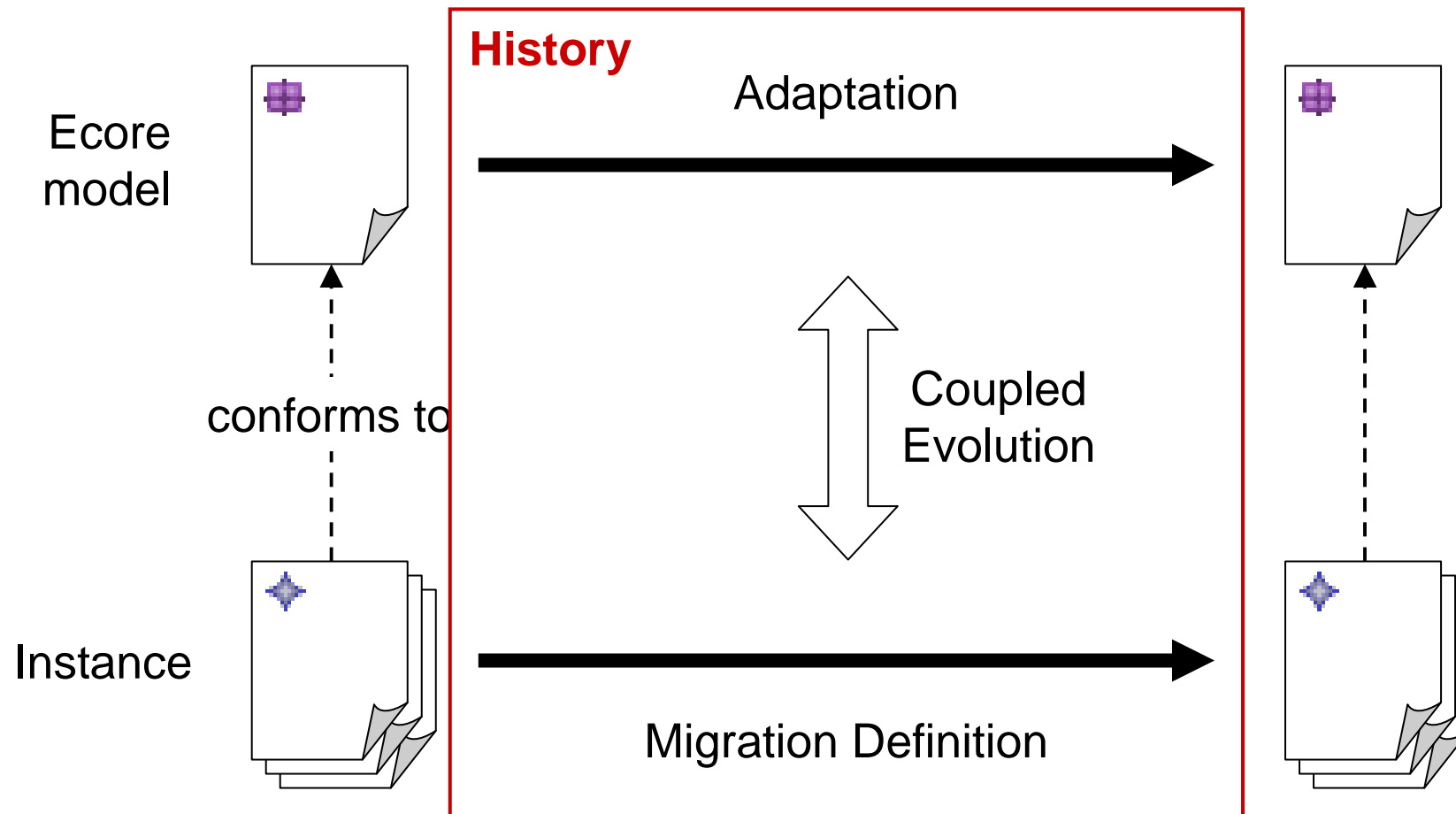
Automated Instance Migration



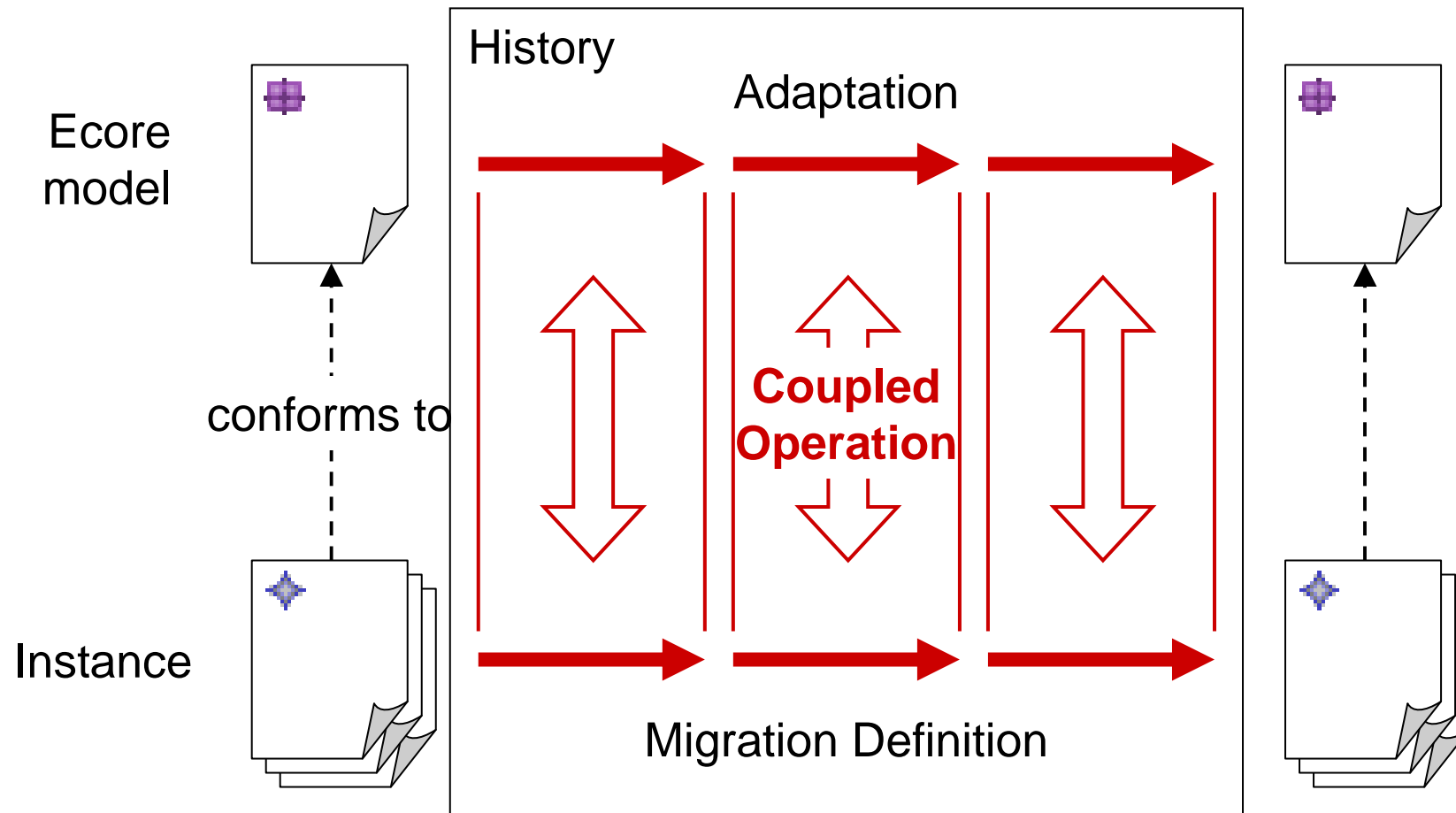
Automated Instance Migration



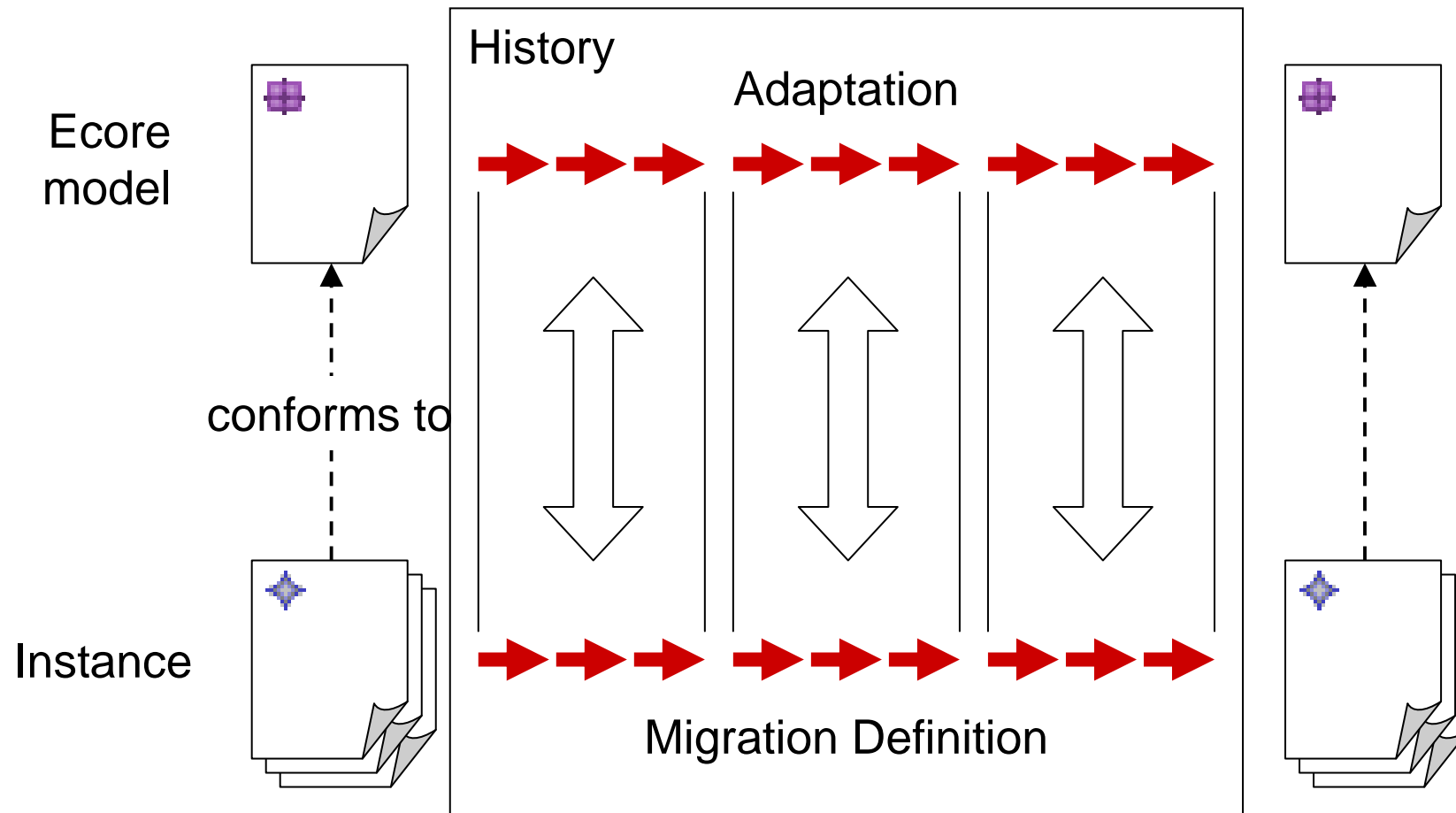
Record the Coupled Evolution



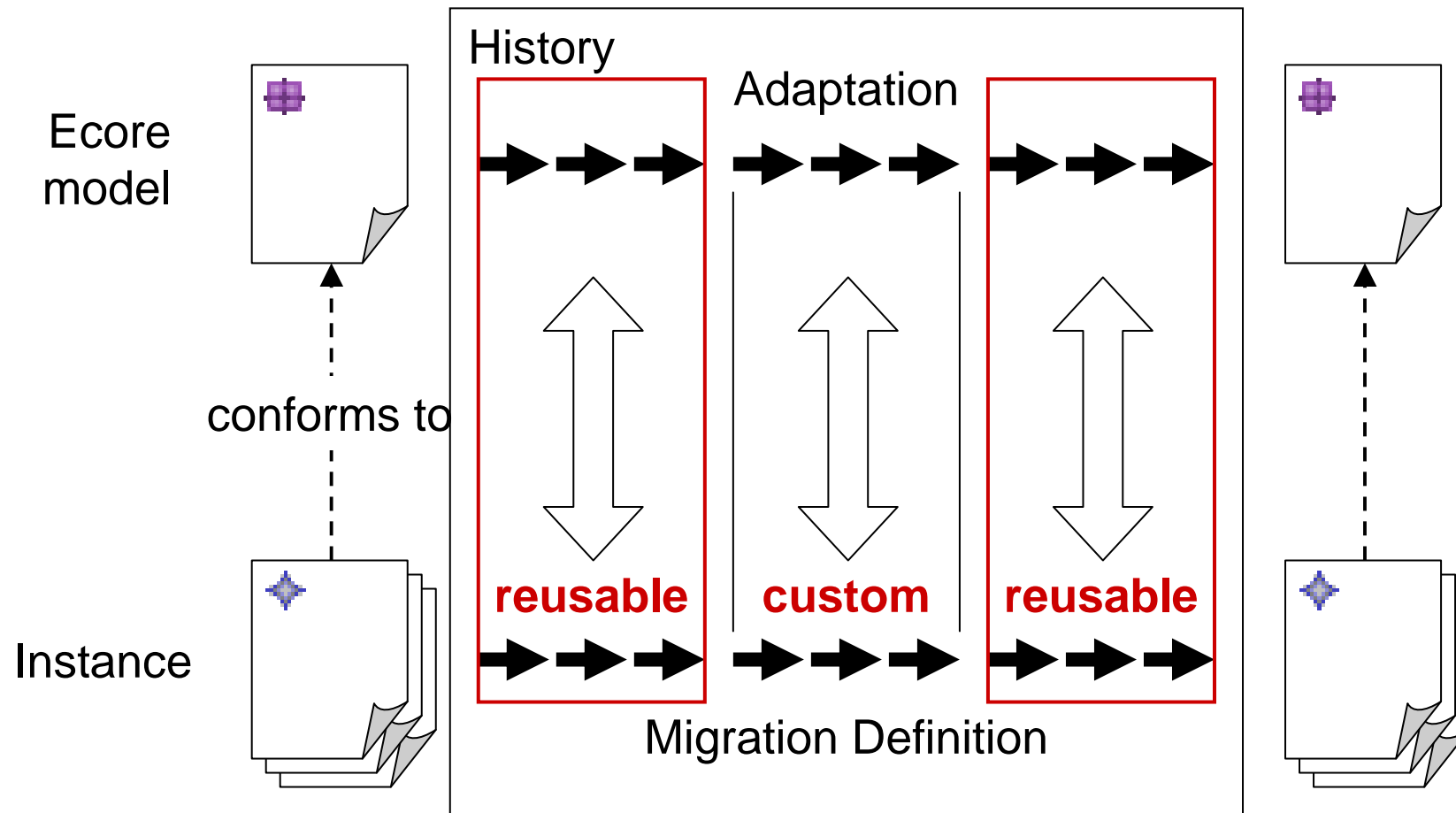
Divide & Conquer



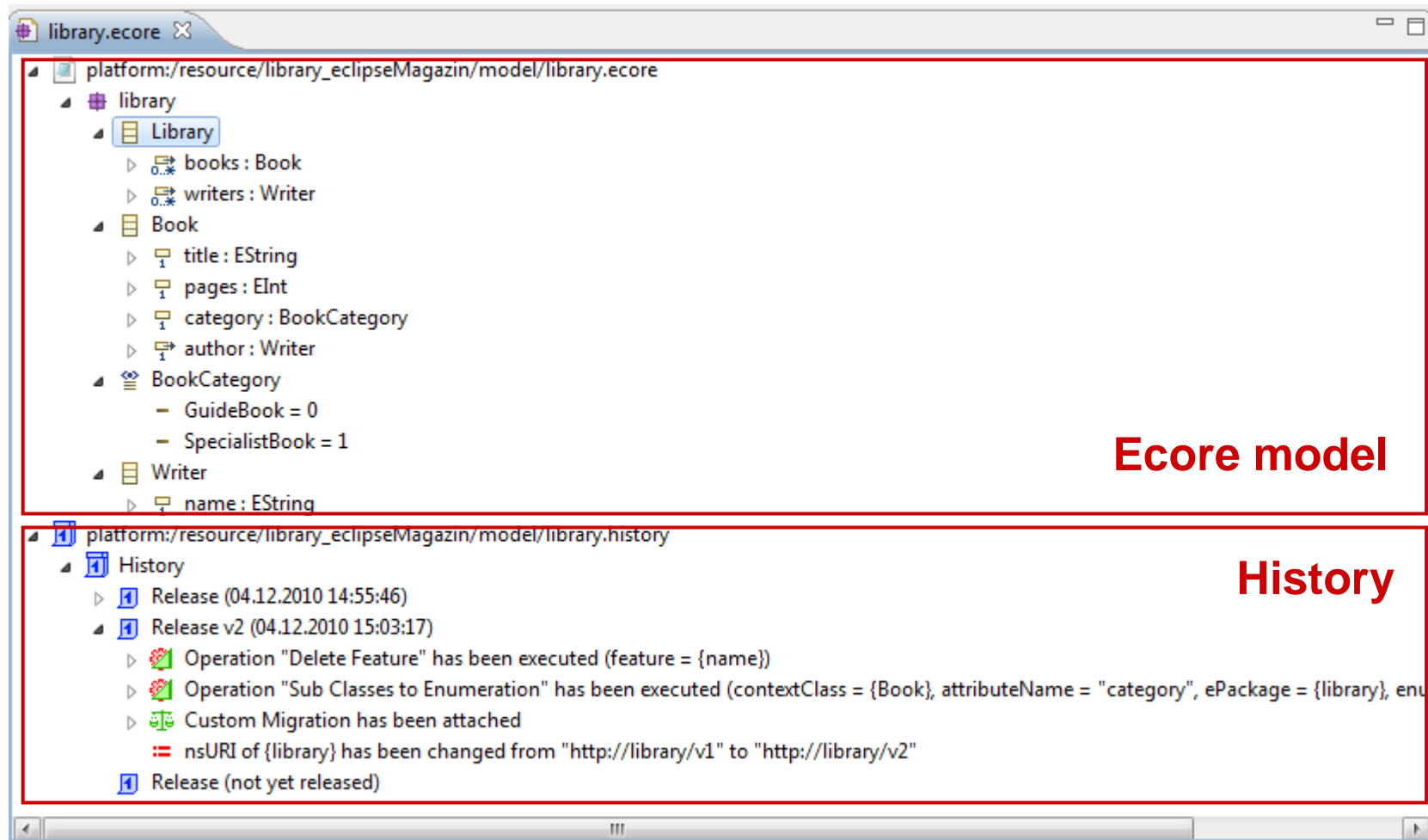
Specify only the Difference



Reuse recurring Coupled Operations



Ecore Model History



The screenshot shows two panels from the Eclipse IDE. The top panel, titled 'library.ecore', displays a tree view of the Ecore model. The bottom panel, titled 'library.history', displays a list of history events.

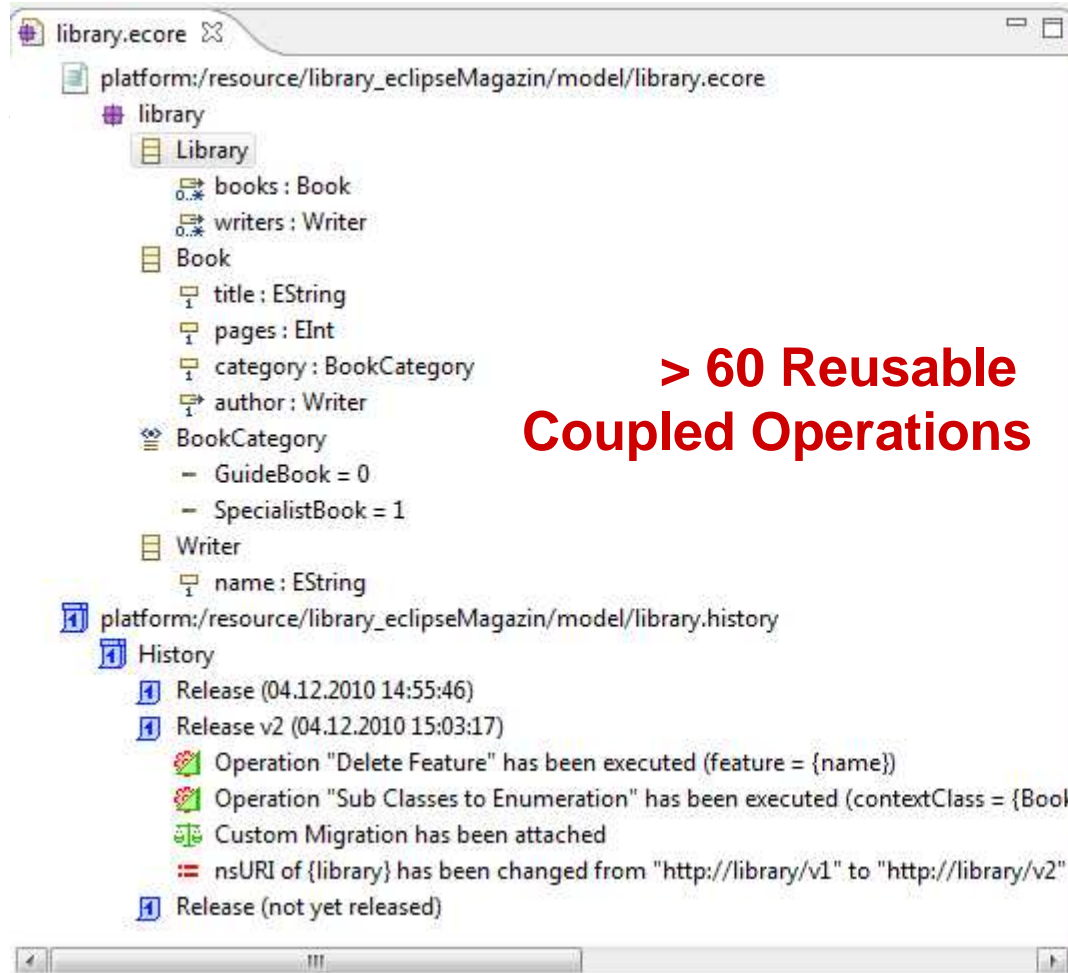
Ecore model

- platform:/resource/library_eclipseMagazin/model/library.ecore
 - library
 - Library
 - books : Book
 - writers : Writer
 - Book
 - title : EString
 - pages : EInt
 - category : BookCategory
 - author : Writer
 - BookCategory
 - GuideBook = 0
 - SpecialistBook = 1
 - Writer
 - name : EString

History

- platform:/resource/library_eclipseMagazin/model/library.history
 - History
 - Release (04.12.2010 14:55:46)
 - Release v2 (04.12.2010 15:03:17)
 - Operation "Delete Feature" has been executed (feature = {name})
 - Operation "Sub Classes to Enumeration" has been executed (contextClass = {Book}, attributeName = "category", ePackage = {library}, enu
 - Custom Migration has been attached
 - nsURI of {library} has been changed from "http://library/v1" to "http://library/v2"
 - Release (not yet released)

Operation Browser



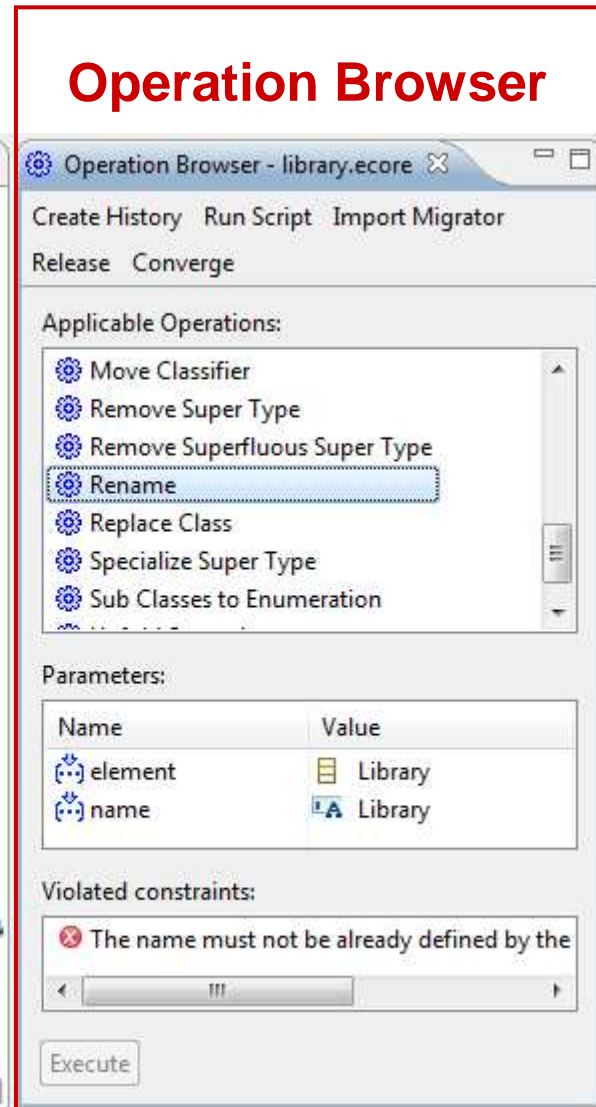
platform:/resource/library_eclipseMagazin/model/library.ecore

- library
 - Library
 - books : Book
 - writers : Writer
 - Book
 - title : EString
 - pages : EInt
 - category : BookCategory
 - author : Writer
 - BookCategory
 - GuideBook = 0
 - SpecialistBook = 1
 - Writer
 - name : EString

platform:/resource/library_eclipseMagazin/model/library.history

- History
 - Release (04.12.2010 14:55:46)
 - Release v2 (04.12.2010 15:03:17)
 - Operation "Delete Feature" has been executed (feature = {name})
 - Operation "Sub Classes to Enumeration" has been executed (contextClass = {Book
 - Custom Migration has been attached
 - nsURI of {library} has been changed from "http://library/v1" to "http://library/v2"
 - Release (not yet released)

**> 60 Reusable
Coupled Operations**



Operation Browser

Create History Run Script Import Migrator
Release Converge

Applicable Operations:

- Move Classifier
- Remove Super Type
- Remove Superfluous Super Type
- Rename**
- Replace Class
- Specialize Super Type
- Sub Classes to Enumeration

Parameters:

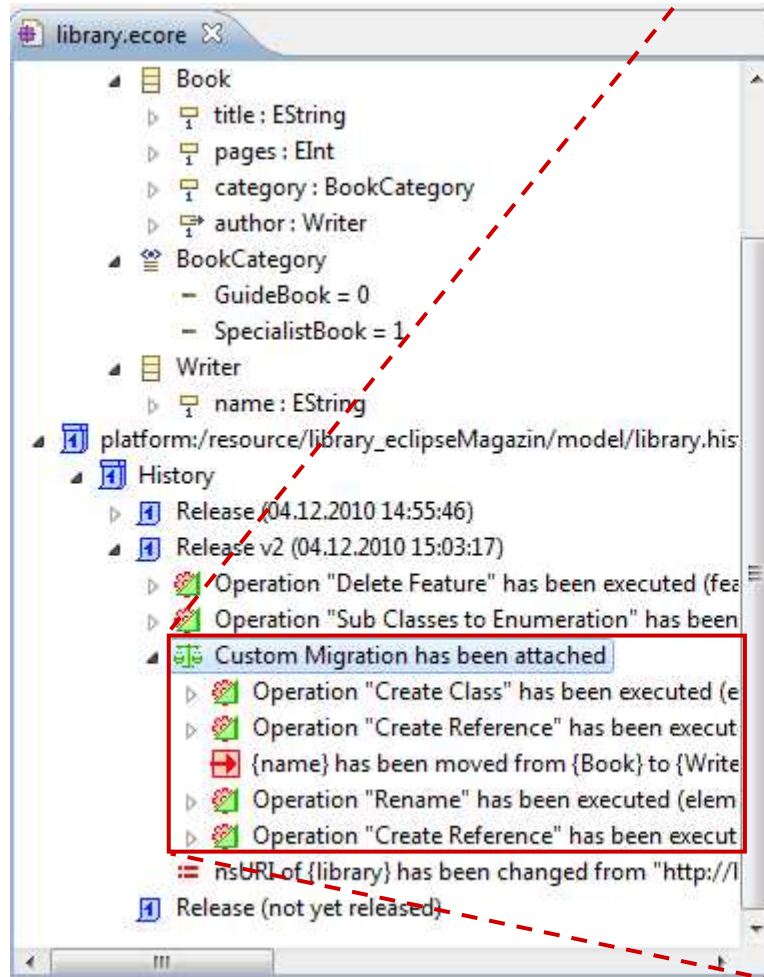
Name	Value
element	Library
name	Library

Violated constraints:

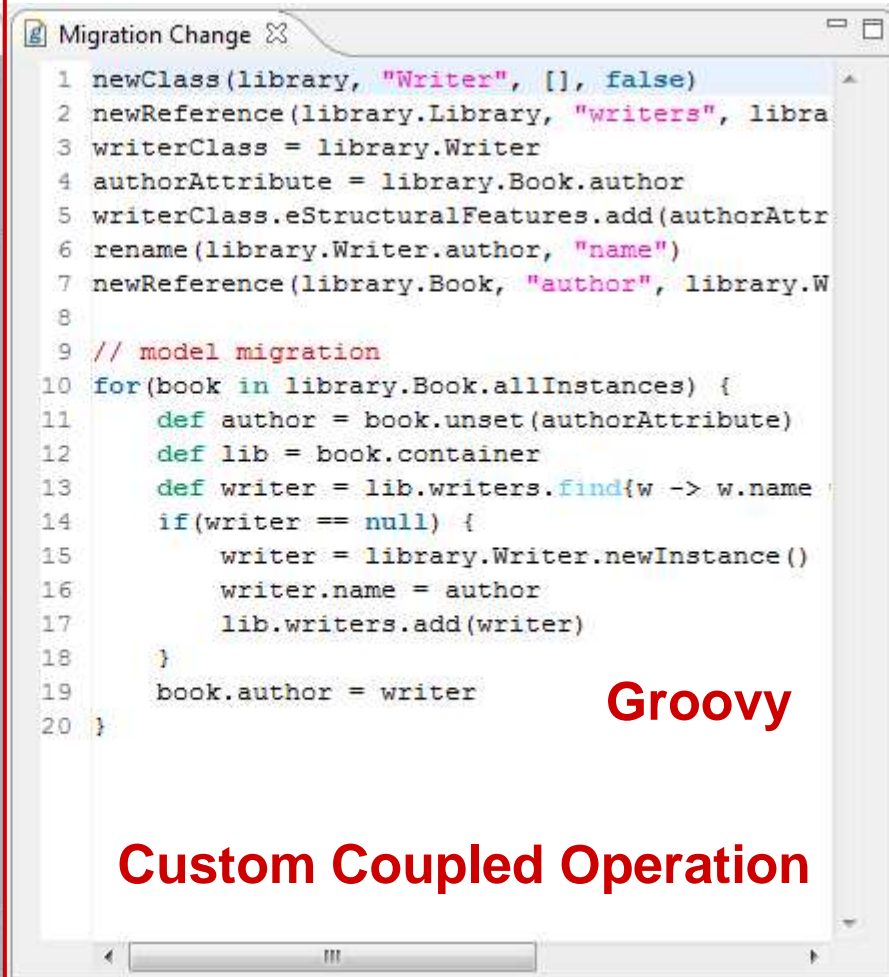
- The name must not be already defined by the

Execute

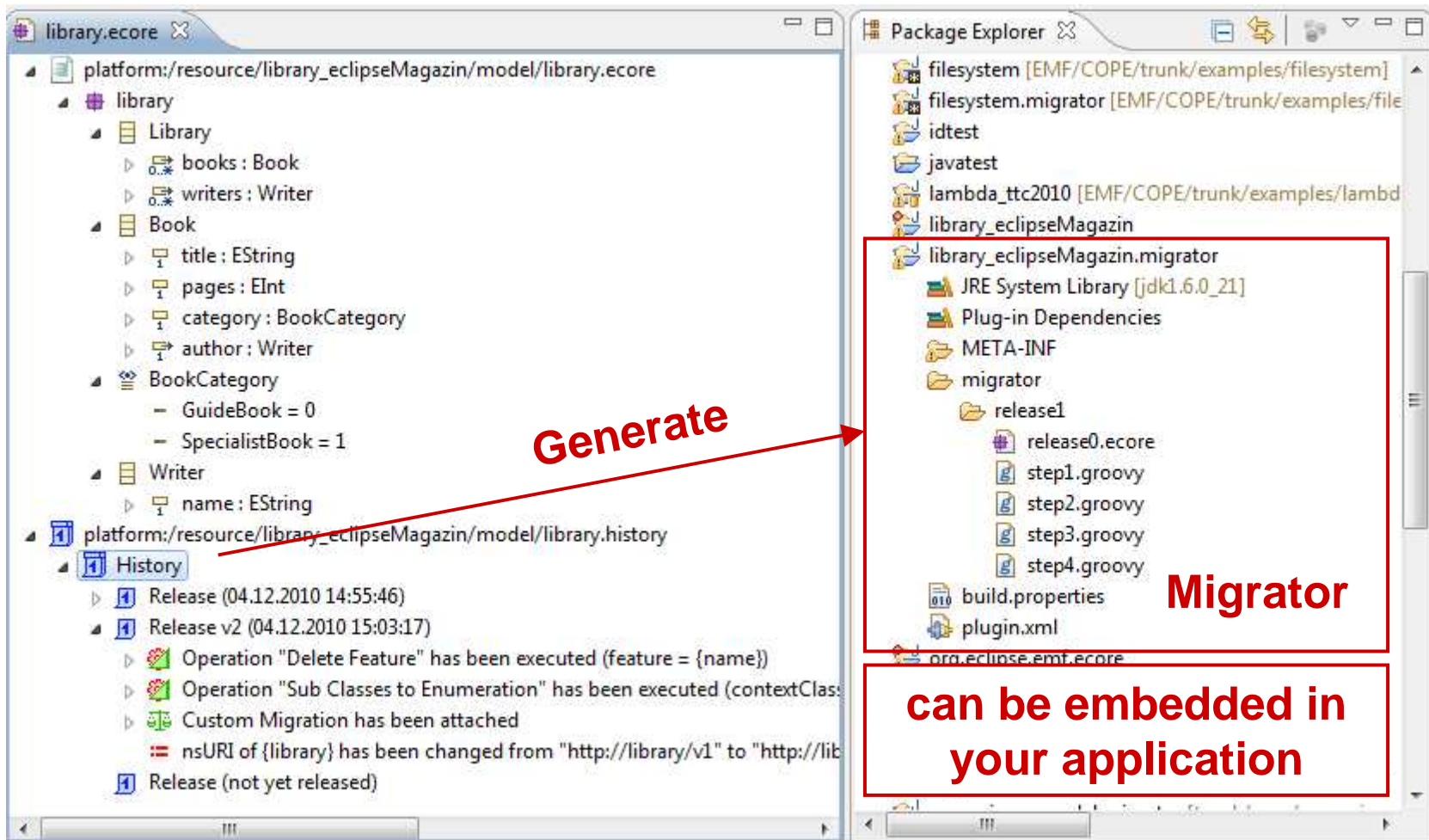
Migration Editor



Migration Editor



Migrator Generation



The screenshot shows the Eclipse IDE interface. On the left, the 'library.ecore' editor displays a UML class diagram with classes 'Library', 'Book', and 'Writer', and their attributes and relationships. Below it, the 'History' view shows a list of operations performed on the model. A red arrow labeled 'Generate' points from the 'History' view to the 'Package Explorer' on the right. The 'Package Explorer' shows a project named 'library_eclipseMagazin.migrator' with a sub-package 'release1' containing files like 'release0.ecore', 'step1.groovy', 'step2.groovy', 'step3.groovy', and 'step4.groovy'. A red box highlights the 'release1' package and its contents, with the word 'Migrator' written in red next to it. Another red box at the bottom of the 'Package Explorer' contains the text 'can be embedded in your application'.

Generate

Migrator

can be embedded in your application

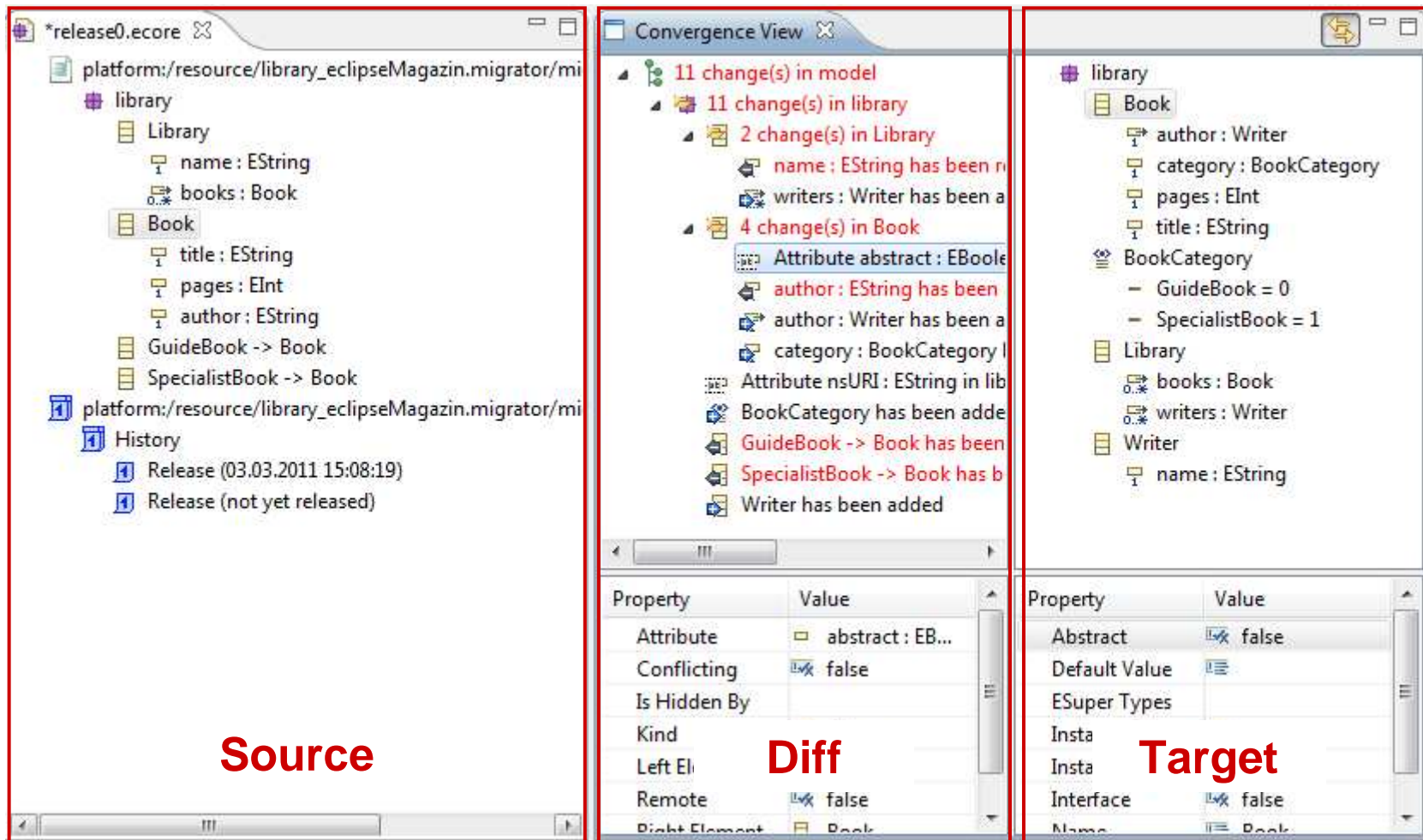
Inspecting the Coupled Evolution

- **Reconstruction:**
restore earlier Ecore model versions
 - **Integrity Check:**
identify problems in the recorded history
 - **History-based Comparison:**
more accurate than EMF Compare
 - **Identify Breaking Changes:**
changes that need a migration attached
-

Refactoring the Coupled Evolution

- **Undoing Changes:**
revert operations that prove to be wrong
 - **Replace Changes with Operation:**
use a reusable coupled operation even though the changes have already been recorded
 - **Reordering Changes:**
put changes together that need a common migration
-

Recovering the Coupled Evolution

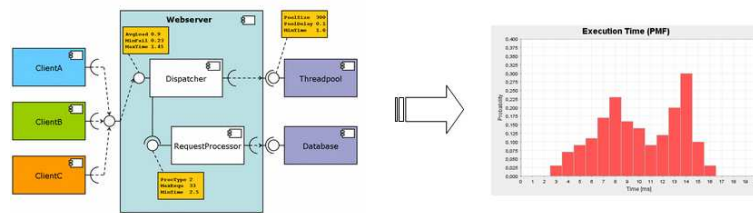


Source

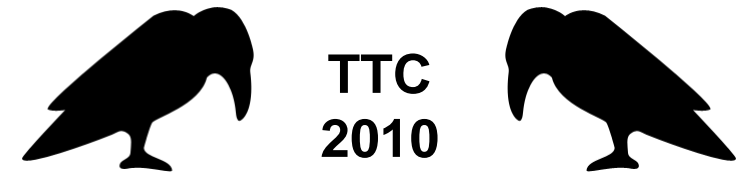
Diff

Target

Case Studies



Palladio Component Model



Transformation Tool Contest

2nd place (of 9)



Graphical Modeling Framework

Projects using Edapt / COPE



Quality Modeling and Controlling



GMF model generator



Unicase
EMFStore

MARINTEK

Norwegian Marine Technology
Research Institute

Summary and Future Work

Benefits of Edapt / COPE

- Automates migration as far as possible
- Preserves intended migration when adapting the Ecore model

Next Steps

- Complete Transition of COPE to Edapt
 - Replace Groovy by Java to define migrations (thanks to Marintek for the support)
-