Operation-based Conflict Detection and Merging

Maximilian Kögel, Jonas Helming, Stephan Seyboth
{koegele, helming, seyboth}@in.tum.de

Chair for Applied Software Engineering
Institut für Informatik
Technische Universität München
State-based vs. Change-based Change Tracking

• State-based:
  – Changes **derived after** they occur
  – Derived by diffing state before and after

• Change-based:
  – Changes **recorded while** they occur
  – No diffing required

• Advantages of change-based:
  – Performance
  – Higher accuracy
  – „Intention“-preservation:
    • Composite changes
    • Time order
Operation-based SCM @ work

• Unicase, a Unified CASE Tool:
  – Requirements
  – Use Cases
  – UML
  – Action Items, Issues and Bug Reports
  – Easily extensible for other models

• Open-Source Eclipse-plugin: http://unicase.org

• Unicase employs operation-based change tracking:
  – on an Eclipse Moding Framework-generated model
  – building on EMF Notification and Resource frameworks
Workspace interaction

Model Change

Commit

Update

Change A  Change B  Change C

Change A  Change B  Change C

Change A  Change B  Change C

Change A  Change B  Change C
Operations Taxonomy (simplified)
Operation-based Version Model

- Version
  - revisions / isVariant
  - createdBy
  - 0..1
- ChangePackage
- ModelState
- Operation
Layers of change

Logical Layer
set of changes comprising logically coherent task

Intentional Layer
Intention of a user changing the model

Model Layer
operations as defined by the model

... model feature A ...

... rename actor generalize class ...

... create class add attribute set subclass ...

17.05.2009
Maximilian Kögel - Operation-based Conflict Detection and Resolution
Operation-based conflict detection

• Conflict detection defines two relations:
  – Requires relation
  – Conflicts relation

• Three conflict severity levels
  – Hard conflict: Applying both changes is not possible
  – Normal conflict: Leads to masking of one change
  – Light conflict:
    • E.g. difference in order of elements
    • can be merged automatically
    • Validation conflicts
Operation-based Merging

- **Input - two lists of operations:**
  - Mine: the local changes to the model
  - Theirs: all changes from base version to update target version

- **Manual merging process:**
  - User selects operation from mine or theirs
  - Result of selection is previewed
  - User confirms selection

- **Output – two lists of operations:**
  - Accepted mine
  - Rejected theirs

- **Merge result:**
  - Reversed rejected theirs followed by accepted mine
Conflicts are often not a technical problem

- Conflicts reflect a different opinion
- Currently conflicts are solved:
  - In Isolation
  - Under time pressure
  - Without discussion

- Conflicts should be discussed:
  - In the team
  - Capturing the results of the discussion
  - Capturing the rationale for a decision
Issue-based Merging

• Rationale Management is successfully employed for software engineering models (IBIS, IOC, RUSE)

• Issue-based conflict resolution reuses issues for merging
  – Issue: conflict in merging
  – Proposals: the conflicting changes
  – Participants: the committers

• Advantages:
  – Conflict resolution can be performed a team activity
  – Real conflict is resolved not a technical problem
Merging Issue Model
Ongoing and Future Work

• Case study on issue-based merging
• Validation conflicts
• Integration with COPE:
  – Coupled Evolution of model and meta-model
  – including (operation-based) history
Backup Slides
### Commit Screenshot

<table>
<thead>
<tr>
<th>Revision</th>
<th>Tag</th>
<th>Date</th>
<th>Author</th>
<th>Log Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>1067</td>
<td>HEAD_BASE</td>
<td>Tue Nov 25 12:22:22 CET 2008</td>
<td>koedel</td>
<td>organized sprint review meeting</td>
</tr>
<tr>
<td>1066</td>
<td></td>
<td>Tue Nov 25 12:18:20 CET 2008</td>
<td>koedel</td>
<td>created new component diagram</td>
</tr>
<tr>
<td>1065</td>
<td></td>
<td>Tue Nov 25 12:10:12 CET 2008</td>
<td>koedel</td>
<td>organized sprint</td>
</tr>
<tr>
<td>1064</td>
<td></td>
<td>Tue Nov 25 12:01:37 CET 2008</td>
<td>koedel</td>
<td>maintained section for otto</td>
</tr>
<tr>
<td>1063</td>
<td></td>
<td>Tue Nov 25 12:01:08 CET 2008</td>
<td>koedel</td>
<td>maintained section for otto</td>
</tr>
<tr>
<td>1062</td>
<td></td>
<td>Tue Nov 25 11:56:59 CET 2008</td>
<td>koedel</td>
<td>maintained section for otto</td>
</tr>
<tr>
<td>1061</td>
<td></td>
<td>Tue Nov 25 11:55:51 CET 2008</td>
<td>koedel</td>
<td>maintained section for otto</td>
</tr>
<tr>
<td>1060</td>
<td></td>
<td>Tue Nov 25 11:50:05 CET 2008</td>
<td>koedel</td>
<td>maintained section for otto</td>
</tr>
<tr>
<td>1059</td>
<td></td>
<td>Tue Nov 25 11:41:35 CET 2008</td>
<td>koedel</td>
<td>maintained section for otto</td>
</tr>
<tr>
<td>1058</td>
<td></td>
<td>Tue Nov 25 11:41:12 CET 2008</td>
<td>koedel</td>
<td>maintained section for otto</td>
</tr>
<tr>
<td>1057</td>
<td></td>
<td>Tue Nov 25 11:39:58 CET 2008</td>
<td>koedel</td>
<td>maintained section for otto</td>
</tr>
<tr>
<td>1056</td>
<td></td>
<td>Tue Nov 25 11:38:24 CET 2008</td>
<td>koedel</td>
<td>maintained section for otto</td>
</tr>
</tbody>
</table>

### Model Element

- Change Package (denglerm@2008-11-25 12:22)
  - Direction of the inheritance arrow
  - The diagram shows strange behaviour
  - Names in title bar of diagram and diagram elements are not correct
  - Cannot Reproduce
  - NullPointer Exception on opening a ComponentService
  - Component diagram service needs
  - Component is shown as service

### Operation

- Log message: 'sorted bugs'
  - Set dueDate attribute
  - Set dueDate attribute
  - Set Status attribute
  - Changed containedWorkItems
  - Set containingWorkpackage
  - Set dueDate attribute
  - Set dueDate attribute
  - Deleted BugReport "Component is shown as service"
Validation Screenshot

<table>
<thead>
<tr>
<th>Severity</th>
<th>Constraint</th>
<th>Description</th>
<th>Affected Model Element</th>
<th>Creator</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ActionItem Assignee Constraint</td>
<td>ActionItem: 'Complete Access control implementation' has no Assignee.</td>
<td>Complete Access control impl.</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ActionItem Activity Constraint</td>
<td>ActionItem: 'Complete Access control implementation' does not have an activity.</td>
<td>Complete Access control impl.</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ActionItem Activity Constraint</td>
<td>ActionItem: 'DiagramEditor save fix' does not have an activity assigned.</td>
<td>DiagramEditor save fix</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ActionItem Activity Constraint</td>
<td>ActionItem: 'Fix transparent overlay icons' does not have an activity assigned.</td>
<td>Fix transparent overlay icons</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ActionItem Activity Constraint</td>
<td>ActionItem: 'Test add items to iteration' does not have an activity assigned.</td>
<td>Test add items to iteration</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ActionItem Activity Constraint</td>
<td>ActionItem: 'Login Dialog' does not have an activity assigned.</td>
<td>Login Dialog</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ActionItem Activity Constraint</td>
<td>ActionItem: 'Implement enum widget' does not have an activity assigned.</td>
<td>Implement enum widget</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ActionItem Activity Constraint</td>
<td>ActionItem: 'Implement and test MEState for opening Links' does not have an activity assigned.</td>
<td>Implement and test MEState for o</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ActionItem Activity Constraint</td>
<td>ActionItem: 'Review Common UI' does not have an activity assigned.</td>
<td>Review Common UI</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ActionItem Activity Constraint</td>
<td>ActionItem: 'Review Navigator' does not have an activity assigned.</td>
<td>Review Navigator</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ActionItem Activity Constraint</td>
<td>ActionItem: 'Implement Enumeration Widget' does not have an activity assigned.</td>
<td>Implement Enumeration Widget</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Class Package Constraint</td>
<td>Class: 'null' is not contained in a package.</td>
<td>unnamedElement</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Class Package Constraint</td>
<td>Class: 'TestClass' is not contained in a package.</td>
<td>TestClass</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Class Package Constraint</td>
<td>Class: 'TestClass1' is not contained in a package.</td>
<td>TestClass1</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Class Package Constraint</td>
<td>Class: 'TestClass2' is not contained in a package.</td>
<td>TestClass2</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Class Package Constraint</td>
<td>Class: 'Class 4' is not contained in a package.</td>
<td>Class 4</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Juran Requirement Constraint</td>
<td>UseCase 'null' is not linked to a functional requirement.</td>
<td>unnamedElement</td>
<td></td>
</tr>
</tbody>
</table>
Merging Screenshot