Synchronisation of data models for collaborative editors

Sebastian Runge
Demo

• Several example applications:
  
  * collaborative ticket system
  * collaborative text editor
  * collaborative storyboard/UML editor
Collaboration concepts

• Pessimistic approach:  
  *locking of central data*

• Optimistic approach:  
  *merging of distributed data*
Basic idea

User A

User B
Basic idea

User A

User B
Basic idea

User A

User B
Basic idea

User A

User B
Basic idea

User A

User B

Sebastian Runge - Synchronisation of data models for collaborative editors
Requirements

• Persistant storage of object structures + reloading

• Replication and synchronisation of object structures (peer to peer)

• Merging and conflict resolution

• “offline mode”
Persistence

• Normally:
  
  *Hibernate, Serialization, XML, …*

• Here:
  
  *record every single change of the model*
  
  *save the model as list of changes*
Detection of model changes

• Design Pattern: Observer/Listener

  • Implementations for common model variants
    (POJOs, Java Beans, EMF, ...)

  • Generation of meta model
    (Fujaba, UML Lab, SDMLib, ...)

• Alternative: differential approach
Synchronisation

- Object management and object IDs
  
  \textit{SDMLib (sdmlib.org)}

- Record every single change of the model

- Transport changes over the network to the other users

- If there are no concurrent changes, the synchronisation is complete
Concurrent changes

• Models “grew apart” and have to be merged

• Possible merge conflicts

• Order of change application is important
Concurrent changes

• Simple solution: sorting and merging of change lists

User Alice
• alice.22
• alice.24
• alice.25

User Bob
• bob.23
• bob.24
• bob.26

Global History
• alice.22
• bob.23
• alice.24
• bob.24
• alice.25
• bob.26

• Conflicts: last edit wins
Concurrent changes

• Problem: deletion of objects

<table>
<thead>
<tr>
<th>Global History</th>
</tr>
</thead>
</table>
| • ... changes ...
| • Alice: delete object O
| • ... changes ...
| • Bob: change object O
| • ... changes ... |
Synchronisation mechanism

Sebastian Runge - Synchronisation of data models for collaborative editors
Result

• Tested distributed storyboard editor with three users in a network

• Editor did not feel slower than a single-user application

• Simple merging solution not suitable for long offline times
Future

- Compression of change lists
- Grouping of changes, transactions
- Undo/Redo
- Synchronisation of array lists
- Resources
Questions?