Saros:
Distributed Project Integration

Benjamin Sypianski, Cedric Laier, Mahmoud Kozae
Knowledge at the beginning
General Idea

Focus on the project’s structure

Important questions:

- How to integrate files between participants of a Saros session?
- How to visualize the whole process to the user?
Project Timeline

APR
Initial Research
Interviewing contributor from the Saros group
Secondary research

MAY
Brainstorming
Taskflow, storyboard
Metaphors, analogies
Paper prototype

JUN
Making real
Technology choice
HiFi prototyping
Formative evaluation

JUL
Finishing
Summative evaluation
Creative Part

- rail network
- puzzle
- power button
- switch
- company fusion
- to unite
- two becomes one
- two streams unite and become a river
- fuse (verschmelzen)
- yellow and red becomes orange
- mixture
- a bag of M&Ms
- something new is created
- copper and tin become bronze
- road block
- east and west germany
Paper Prototype & First Iterations

Paper Prototypes

[Diagram showing a moderator, user, and 'computer' with 'zuschauer' nearby]
Biggest Challenges

- Find a way to visually compare projects
- Identify possible file & folder states
- Find the "right" icons and data visualisation to transfer this knowledge to the user
- Come up with a task flow to resolve conflicts
Hi-Fi Prototype

LIVE DEMO

also hosted on a S3 Bucket here
Testing the Hi-Fi Prototype

System Usability Scale (SUS)

- We wanted to get an high level overview of the view of usability for the user
- Should allow us to quickly compare the effect of adjustments
- Tested professional Software-Engineers
- Overall average Score of 75,6 Points (> 71,4 = Good ; > 85,5 = Excellent)
## Outlook (Eclipse)

![Eclipse IDE](image)

<table>
<thead>
<tr>
<th>Merge Actions - Cheatsheet</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Responsibilities of merge target:</strong></td>
</tr>
<tr>
<td>No action required</td>
</tr>
<tr>
<td>Automatic merge</td>
</tr>
<tr>
<td>Manual merge required</td>
</tr>
<tr>
<td><strong>When merge to Target:</strong></td>
</tr>
<tr>
<td>Copy to right</td>
</tr>
<tr>
<td>Delete right</td>
</tr>
<tr>
<td><strong>When merge to Source:</strong></td>
</tr>
<tr>
<td>Copy to left</td>
</tr>
<tr>
<td>Delete left</td>
</tr>
</tbody>
</table>

**Merge Projects**

- **Name**: js.js, index.js, index.css, app.js, app.css, node_modules
- **Size**: 10, 20, 30
- **Modified**: true, false

- **Merge to**: Base, External, New
- **Actions**: Keep left, Keep right, Delete left, Delete right, Take base, Resolve Conflict

---

*Note: The images and diagrams are placeholders.*
Outlook (VS Code)
Outlook

What’s missing:

- Add more Scenarios: Merge base into external project & Merge base project + external project into new project
- Extend Prototype and have more Test-Iterations
- Finetune Icon mapping
- Add color-codes for folder/file states
- Extend the prototype by a “Resolve Conflicts” View
Thank you!

All artefacts can be found here:
Images, Sheets, Assignments, Tests etc.: https://drive.google.com/drive/u/0/folders/1cKRxeE958sb23oDIQd008UV9nmzOWXYwX
Web Hi-Fi Prototype Code: https://github.com/Rintel/UCD-HiFi-Prototype
MacOS Hi-Fi Prototype Code: https://github.com/nepysb/SarosUCDMacOSPrototype