Evaluating Tools that Support Pair Programming in a Distributed Engineering Environment

Dietmar Winkler  Stefan Biffl  Andreas Kaltenbach

Institute of Software Technology and Interactive Systems,
Vienna University of Technology

dietmar.winkler@tuwien.ac.at
http://qse ifs.tuwien.ac.at
Motivation

Challenges in modern software development practices:
- Delivery of high-quality software products within short iterations.
- Ability to respond to frequent and changing requirements.
- Selection of best-practice tools for project application.

Manage these challenges …
- Ongoing global software development (24h development).
- Application of agile and flexible software processes for project planning, monitoring and control.
- Application of established (agile) practices, e.g., pair programming.

→ Distributed pair programming
→ Tool support to enable distributed and collaborative software development.

Goals:
- Elicitation of basic requirements as foundation for distributed pair programming (DPP) application.
- Developing an evaluation framework for efficient tool evaluation.
- Application of the evaluation framework to identify tools for DPP support.
Traditional Pair Programming

- Pair programming is an established agile practice for efficient software development.
- Ability to support code construction, design, and test ("pair activities" and "pair tasks").

- Basic idea of pair programming:
  - 2 developers (a pair) elaborate on a software artifact concurrently sharing a common working (co-located) environment (screen, keyboard, and mouse).
  - Clearly defined roles and change of role assignments:
    Driver (implementation) and Observer / Navigator (e.g., continuous reviews, provides)

- Benefits of pair programming (derived from various empirical studies):
  - Increased quality, effectiveness, and productivity.
  - Improved team communication.
  - Focus on a common artifact ("Pair Pressure")
  - Pair learning.

- Is Pair Programming applicable in a distributed environment?
Distributed Pair Programming

- Applying traditional pair programming requires a shared work space in a co-located environment.

- Basic idea of Distributed Pair Programming (DPP):
  - Pair programming and a shared workspace over distances.
  - Gaining benefits of traditional pair programming.
  - Continuous collaboration.

- Basic pre-conditions:
  - Efficient communication and collaboration mechanisms (e.g., screen sharing, communication channels, and gesturing).
  - Efficient data exchange approaches.
  - Workspace control and awareness (participants, artifacts, tasks).
  - Floor control (ability to change of roles and trace changes)
  - Tools that support collaboration and continuous interaction within a common working environment to bridge geographical and temporal distances.

Process for Tool Selection

Four basic steps for tool selection [Poston, 1992]:

1. Analysis and classification of user requirements and expected tool properties.
2. Elicitation and prioritization of selection criteria.
3. Classification of candidate tools.
4. Assessment of tools according to a pre-defined evaluation scheme.
Step 1: Requirements: Analysis of User Requirements

- Systematic research of existing analysis results regarding the application domain.

- Brainstorming of related stakeholders to capture and complete individual (tool) requirements.

- Samples of important basic requirements categories for DPP supporting tools:
  - Workspace control and awareness. Visibility of participants within the working environment; defined mouse and keyboard control.
  - Screen Sharing support.
  - Floor control. Transparent and traceable changes within an artifact by roles.
  - Gesturing. Ability to point to specific aspects of interest (e.g., defects) by using a second pointer device.
  - Efficient information exchange to support communication and collaboration, e.g., textual and/or voice chat, video conferencing.
  - Platform-independence, usability, tool documentation.
Step 2: Requirements: Classification and Prioritization

- Elicitation, classification and prioritization of collected requirements
- Requirements Elicitation workshop (EasyWinWin) according to Boehm et al, 2001.

- Snapshot of collected requirements.

<table>
<thead>
<tr>
<th>General tool Properties (8)</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Requirements (2)</td>
<td>Priority</td>
</tr>
<tr>
<td>Support of category supported features (e.g., Screen Sharing, Collaborative Work Support)</td>
<td>Critical (C)</td>
</tr>
<tr>
<td>Support of Workspace Awareness</td>
<td>Critical (C)</td>
</tr>
<tr>
<td>Floor Control (7)</td>
<td>Priority</td>
</tr>
<tr>
<td>Support of Floor Control</td>
<td>Critical (C)</td>
</tr>
<tr>
<td>Support of Role Changes</td>
<td>Critical (C)</td>
</tr>
<tr>
<td>Role Assignment information</td>
<td>Medium (M)</td>
</tr>
<tr>
<td>Gesturing (4)</td>
<td>Priority</td>
</tr>
<tr>
<td>Second Pointer for the Navigator</td>
<td>Critical (C)</td>
</tr>
<tr>
<td>Support of Highlighting</td>
<td>Low (L)</td>
</tr>
<tr>
<td>Communication (5)</td>
<td>Priority</td>
</tr>
<tr>
<td>Voice Channels</td>
<td>High (H)</td>
</tr>
<tr>
<td>Textual Chat</td>
<td>Medium (M)</td>
</tr>
<tr>
<td>Video channel</td>
<td>Low (L)</td>
</tr>
<tr>
<td>Platform Independence (3)</td>
<td>Priority</td>
</tr>
<tr>
<td>Usability (10)</td>
<td>Priority</td>
</tr>
</tbody>
</table>

Requirements prioritization

<table>
<thead>
<tr>
<th>Prioritization</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>10</td>
</tr>
<tr>
<td>High importance</td>
<td>5</td>
</tr>
<tr>
<td>Medium importance</td>
<td>2,5</td>
</tr>
<tr>
<td>Low importance</td>
<td>1</td>
</tr>
</tbody>
</table>

- **Critical**: basic functions for DPP.
- **High importance**: Pre-conditions for efficient PP application.
- **Medium importance**: Requirements and attributes that can increase tool application.
- **Low importance**: Nice-to-have features.
Step 3: Candidate Tools: Search for Tools and Classification

Search for Candidate Tools:

- Identification of tools applicable for Computer Supported Collaborative Work:
  - Based on requirements categories.
  - Tools to support distributed collaborative work (independent of DPP).
  - Tools designed for DPP support.

Basic Classification of Candidate Tools.

- Screen-Sharing Applications.
  - Screen-Sharing without Interaction: Exchanging screen content (e.g., VNC) typically used by system administrators.
  - Screen-Sharing with Interaction: Additional features like Whiteboards, Chat (e.g., MS Netmeeting). Typical used for video conferencing.
  - Tools for explicit DPP support

- Collaboration-aware applications.
  - Distributed editors
  - Integrated Development Environments (IDE).
Snapshot of Candidate Tools

- Sample snapshot of candidate tools depending on the availability of tools.

**Screen-Sharing Applications**

- Without Interaction:
  - VNC
  - Hanks VNC extension
  - Famatech Remote Administrator
  - Symantec pcAnywhere

- With Interaction:
  - Microsoft Netmeeting
  - Marratech
  - Milos ASE System

**Collaborative Work Support**

- Collaborative Editors:
  - ACE
  - Gobby
  - GrewpEdit
  - MoonEdit
  - SynchroEdit
  - TalkAndWrite (Skype)

- IDEs:
  - Borland CodeWright
  - Borland Jbuilder
  - DocSynch
  - Collab.NetBeans

- **DPP Designed:**
  - Copper
  - SubEthaEdit
  - TUKAN

- DPP Designed:
  - Distr. XP Support Tool
  - Moombas MCIDE
  - PEP
  - Sangam
  - Saros
  - XelicP
  - XPairtise
Step 4: Tool Assessment: Evaluation Framework

- Evaluation framework for systematically assessing candidate tools with respect to classified requirements.

- 4b. Evaluation framework definition.
- 4c. Evaluation and assessment of tools based on captured scenarios.
Step 4a: Tool Assessment: Identification of Success-Critical Scenarios

Scenarios:

- Typical workflows and tasks based on user requirements.
- Guidelines for real-world tasks
- Scenario brainstorming workshop for DPP application

Selection of 6 basic scenarios:

- General scenarios (applicable to various types of tools):
  - Tool installation and configuration.
  - Tool performance.
- DPP specific scenarios
  - Initialization of a DPP session
  - Support of role assignment changes (Floor control).
  - Session Management (Storing/Restoring sessions).
  - Sample application for executing a small implementation task.

- Tool evaluation is based on scenario application and a subjective assessment of the Pair Programming Pair (Team result after discussion).
Step 4b: Tool Assessment: Evaluation Framework Definition

- Classified requirements and priorities (y-axis)
- Candidate tools for planned evaluation (x-axis)
- Tool evaluation matrix:
  - Estimation of the degree of requirements coverage by the tool:
    - Likert-scale: 0.. not supported / 5 ... fully supported.
    - 0/1 estimation if applicable, e.g., support of an individual platform (yes/no).
  - Weighting of the subjective assessment (acc. to requirement prioritization)
Step 4c: Tool Assessment:
Snapshot of the Evaluation Matrix

- Tool application based on defined scenarios executed by real distributed pairs.
- Subjective Team assessment during/after tool application based on the classified requirements according to the evaluation framework.

<table>
<thead>
<tr>
<th>General tool Properties (8)</th>
<th>General Requirements (2)</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>Support of category supported features</td>
<td>Critical (C)</td>
<td>4</td>
</tr>
<tr>
<td>Support of Workspace Awareness</td>
<td>Critical (C)</td>
<td>5</td>
</tr>
</tbody>
</table>

| Floor Control (7) | Support of Floor Control | Critical (C) | 0 | 0 |
| Support of Role Changes | Critical (C) | 0 | 0 |
| Role Assignment information | Medium (M) | 0 | 0 |

| Gesturing (4) | Second Pointer for the Navigator | Critical (C) | 1 | 1 |
| Support of Highlighting | Low (L) | 5 | 25 |

| Kommunikation (5) | Voice Channels | High (H) | 0 | 0 |
| Textual Chat | Medium (M) | 0 | 0 |
| Video channel | Low (L) | 0 | 0 |

| Platform (3) | Usability (10) | Rate | Weighted | Rate | Weighted | Rate | Weighted | Rate | Weighted |
| Screen Sharing | VNC | Netmeeting | Collaborative Work | Copper | PEP |
| Rate | Weighted | Rate | Weighted | Rate | Weighted |
|----------------------------|----------------|----------------|----------------------------|----------------|----------------|----------------------------|----------------|----------------|----------------------------|
| VNC | 4 | 40 | 4 | 40 | 4 | 40 | 3 | 30 |
| Netmeeting | 5 | 25 | 4 | 40 | 4 | 40 | 2 | 20 |
| Copper | 0 | 0 | 3 | 30 | 5 | 50 | 3 | 30 |
| PEP | 0 | 0 | 2 | 20 | 5 | 50 | 4 | 40 |
| Medium (M) | 0 | 0 | 2 | 5 | 5 | 12.5 | 4 | 10 |

- Second Pointer for the Navigator: Critical (C) - Rate 1, Weighted 1
- Support of Highlighting: Low (L) - Rate 5, Weighted 25
- Voice Channels: High (H) - Rate 0, Weighted 25
- Textual Chat: Medium (M) - Rate 0, Weighted 10
- Video channel: Low (L) - Rate 0, Weighted 5
Step 4c: Tool Assessment: Aggregating Evaluation Results

- Aggregation of individual tool results to requirements categories (summarizing individual ratings).
- Threshold of a 3-level assessment based on the maximum value per requirements category.
  - Little tool support of attribute/requirement: 0-33% (red marked).
  - Medium support (33-66%) (orange marked).
  - Comprehensive support: 66-100% (green marked).

<table>
<thead>
<tr>
<th>General Tool Properties (8)</th>
<th>General Requirements (2)</th>
<th>Floor Control (7)</th>
<th>Gesturing (4)</th>
<th>Kommunikation (5)</th>
<th>Plattform (3)</th>
<th>Usability (10)</th>
<th>Total (39)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rate</td>
<td>Weighted</td>
<td>Rate</td>
<td>Weighted</td>
<td>Rate</td>
<td>Weighted</td>
<td>Rate</td>
<td>Weighted</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>7</td>
<td>7</td>
<td>11</td>
<td>11</td>
<td>6</td>
<td>5</td>
</tr>
</tbody>
</table>
Results:
Evaluation of Selected Candidate Tools

- **Share** of requirements coverage (per category) based on the weighted assessment results.

<table>
<thead>
<tr>
<th>Tool</th>
<th>Screen-Sharing</th>
<th>Collaboration</th>
<th>Basics</th>
<th>Gesturing</th>
<th>Floor Control</th>
<th>Communication</th>
<th>Platform</th>
<th>Usability</th>
<th>Tool</th>
<th>Rate</th>
<th>Weighted</th>
<th>Weighted [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Xpaitise</td>
<td>X</td>
<td></td>
<td>100%</td>
<td>22%</td>
<td>82%</td>
<td>38%</td>
<td>100%</td>
<td>83%</td>
<td>56%</td>
<td>100</td>
<td>471</td>
<td>72%</td>
</tr>
<tr>
<td>Copper</td>
<td>X</td>
<td></td>
<td>80%</td>
<td>11%</td>
<td>89%</td>
<td>29%</td>
<td>100%</td>
<td>83%</td>
<td>31%</td>
<td>85</td>
<td>442</td>
<td>68%</td>
</tr>
<tr>
<td>TalkAndWrite</td>
<td>X</td>
<td></td>
<td>90%</td>
<td>85%</td>
<td>0%</td>
<td>83%</td>
<td>42%</td>
<td>73%</td>
<td>75%</td>
<td>91</td>
<td>396</td>
<td>61%</td>
</tr>
<tr>
<td>Net-Meeting</td>
<td></td>
<td>X</td>
<td>80%</td>
<td>30%</td>
<td>42%</td>
<td>83%</td>
<td>42%</td>
<td>69%</td>
<td>69%</td>
<td>87</td>
<td>390</td>
<td>60%</td>
</tr>
<tr>
<td>PEP</td>
<td>X</td>
<td></td>
<td>50%</td>
<td>5%</td>
<td>66%</td>
<td>29%</td>
<td>100%</td>
<td>73%</td>
<td>50%</td>
<td>79</td>
<td>351</td>
<td>54%</td>
</tr>
<tr>
<td>DocSync</td>
<td>X</td>
<td></td>
<td>80%</td>
<td>0%</td>
<td>63%</td>
<td>24%</td>
<td>100%</td>
<td>60%</td>
<td>56%</td>
<td>72</td>
<td>342</td>
<td>52%</td>
</tr>
<tr>
<td>VNC</td>
<td></td>
<td>X</td>
<td>90%</td>
<td>41%</td>
<td>11%</td>
<td>0%</td>
<td>83%</td>
<td>76%</td>
<td>44%</td>
<td>69</td>
<td>328</td>
<td>50%</td>
</tr>
<tr>
<td>Gobby</td>
<td>X</td>
<td></td>
<td>90%</td>
<td>5%</td>
<td>0%</td>
<td>21%</td>
<td>100%</td>
<td>82%</td>
<td>50%</td>
<td>61</td>
<td>304</td>
<td>47%</td>
</tr>
<tr>
<td>GrewpEdit</td>
<td>X</td>
<td></td>
<td>70%</td>
<td>16%</td>
<td>0%</td>
<td>25%</td>
<td>100%</td>
<td>78%</td>
<td>25%</td>
<td>56</td>
<td>284</td>
<td>44%</td>
</tr>
<tr>
<td>MoonEdit</td>
<td>X</td>
<td></td>
<td>60%</td>
<td>16%</td>
<td>0%</td>
<td>0%</td>
<td>100%</td>
<td>86%</td>
<td>56%</td>
<td>58</td>
<td>282</td>
<td>43%</td>
</tr>
<tr>
<td>NetBeans Coll.</td>
<td>X</td>
<td></td>
<td>60%</td>
<td>0%</td>
<td>8%</td>
<td>29%</td>
<td>100%</td>
<td>63%</td>
<td>81%</td>
<td>68</td>
<td>249</td>
<td>38%</td>
</tr>
<tr>
<td>ACE</td>
<td>X</td>
<td></td>
<td>50%</td>
<td>22%</td>
<td>8%</td>
<td>8%</td>
<td>83%</td>
<td>71%</td>
<td>75%</td>
<td>57</td>
<td>250</td>
<td>38%</td>
</tr>
</tbody>
</table>

Maximum reachable Assessment values: 159  652
Conclusion & Further Work

- Applying DPP requires a strong tool support with focus on specific requirements for communication, interaction, data exchange, and collaboration.
- **Evaluation frameworks**, considering requirements classes, tool categories, and scenarios can help to assess candidate tools systematically.
- The **results** showed strengths and weaknesses and can be the starting point for further development of tools to efficiently support DPP in a distributed environment.
  - No tool under investigation supported DPP without limitation.
  - Strong benefits for the top-2 tools, especially designed for DPP.
  - Screen-sharing application can also support DPP to some extent.
- The proposed evaluation framework can support
  - **Project managers and developers** in selecting appropriate tool for project application.
  - **Tool vendors** to identify improvement options for DPP.

- **Future work** includes
  - Improvement and evaluation of the proposed process, i.e., refinement of requirements.
  - Extending the number of tools (including commercial tools).
  - Pilot application of most promising tools in real world project to get feedback from industry on the evaluation framework.
Evaluating Tools that Support Pair Programming in a Distributed Engineering Environment

Dietmar Winkler, Stefan Biffl, Andreas Kaltenbach
Vienna University of Technology
Institute of Software Technology and Interactive Systems
http://qse ifs.tuwien.ac.at
Dietmar.Winkler@tuwien.ac.at