Introducing FastFix

Emitzá Guzmán, Technische Universität München
CREST workshop
May 2011
Software engineers spend more than 50% of their time working on software maintenance tasks (Fjeldstad and Hamlen)
Maintenance

Software maintenance is a key factor to customer’s perception of software product quality.
Maintenance constitutes a growing percentage of the software business market.

- Annual maintenance of software products tends to range from 15% to 25% of the software license list prices (Disbrow, and Igou)
FastFix

• June 2010 – November 2012
• Participants

<table>
<thead>
<tr>
<th>Participant</th>
<th>Country</th>
</tr>
</thead>
<tbody>
<tr>
<td>S2</td>
<td>Spain</td>
</tr>
<tr>
<td>Technische Universität München</td>
<td>Germany</td>
</tr>
<tr>
<td>INESC-ID</td>
<td>Portugal</td>
</tr>
<tr>
<td>Lero</td>
<td>Ireland</td>
</tr>
<tr>
<td>Prodevelop</td>
<td>Spain</td>
</tr>
<tr>
<td>TXT</td>
<td>Italy</td>
</tr>
</tbody>
</table>
FastFix

• **Main Goal:** Provide maintenance services to software applications that are:

  – Time efficient
  – Low cost
  – Strongly accurate
How

• Collect information about application execution and user interaction
• Use collected information to:
  – Detect faulty situations (execution errors, performance degradation
  – Possible causes for faulty situations
• Support automatic fault replication
• Generate and deploy patches
How: Error Handling Strategies

- **Recordable symptoms**: Recommend error report
- **Reproducible error**: Replicate fault
- **Known causes**: Show context information, Recommend causes to engineer, Inform user about error causes
- **Known solution**: Recommend solution to engineer, Recommend fix to user
- **Automatically resolvable**: Generate Patch
- **Self-heal**: Introducing FastFix
Description: App crashes during save operation.

ID: 155362

Error Type: Unhandled Exception

Status: Open

Owner: Homer Simpson

Priority: High

Reporter: FastFix
Introducing FastFix

**Possible Causes:**

<table>
<thead>
<tr>
<th>Cause</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unhandled Exception in HelloWorld</td>
<td>98%</td>
</tr>
<tr>
<td>Usability failure</td>
<td>45%</td>
</tr>
<tr>
<td>Configuration error</td>
<td>12%</td>
</tr>
</tbody>
</table>

**Possible Solutions to Unhandled Exception in HelloWorld:**

<table>
<thead>
<tr>
<th>Solution</th>
<th>Success Probability</th>
<th>Worked for</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disable part of functionality as emergency fix.</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>Change app configuration</td>
<td>76%</td>
<td>89%</td>
</tr>
</tbody>
</table>
Introducing FastFix
Research Lines

- Event Correlation
- Context Elicitation and User Modelling
- Patch Generation
- Performance Security and Privacy
- Fault Replication
FastFix Overall Architecture

- Runtime Environment
- Operating System
- Software Application
- FastFix Client
  - Reporting Interface
  - User Profile
- Communication Infrastructure
- FastFix Server
  - Communication Infrastructure
  - Patch Generator
  - Event Correlator
  - Fault Replicator
- Development Environment
- Test Environment

Legend:
- Blue: FastFix component
- Grey: External component

Introducing FastFix
20.05.2011
What does all this have to do with search-based software engineering?
• Currently search-based techniques are not being used in FastFix.

• Some possible applications could help answer the following questions:
  – Which code blocks give more information concerning a certain feature or are more relevant for a programmer executing a certain task (e.g. fixing a bug)?

  – What is the best way to fix a certain bug? (self-healing) / What is the best patch?

  – What is the best allocation of human resources for a task? (automatically assign developers to a bug based on previous software contributions)
Contact: emitza.guzman@mytum.de