A Tool for Managing Software Architecture Knowledge

M. Ali Babar¹ & Ian Gorton²
¹Lero, University of Limerick, Ireland
²Data Intensive Computing, PNNL, USA

Architecture Knowledge Management Issues

- Unavailability of architecture design knowledge
  - System evolution becomes hard
  - Difficult to identify design errors
- Use of COTS without fully understanding the assumptions
- Situation is worst in case of FLOSS components
- Lack of support in terms of practices and tools
  - What types of architecture knowledge are useful?
  - How to store and manage the knowledge?
  - Make knowledge capture cost-effective
Capturing architecture knowledge

- An architecture embodies crucial design decisions
  - Rarely captured in architecture docs (if they exist!)
- If rationale behind design decisions is lost:
  - System evolution becomes hard
  - Difficult to identify design errors
- We did a survey:
  - 80% can’t understand designs without adequate docs
  - 73% forget why they designed something!
  - Impediments to capturing design info:
    - 61% have no time/budget/tools
- And built an architecture knowledge management tool - BRedB

Tool for Managing Architecture Knowledge

- PAKME: A Process-based Software Architecture Knowledge Management Infrastructure
  - A model of architecture knowledge management
  - Templates for capturing and structuring architecture knowledge
  - Repository of architectural knowledge and experiences
  - Resource for software architects
  - Source of adaptive software architecture processes
  - Support for owning technical and process knowledge
A partial model of Architecture Knowledge

---

Component View of PAKME Architecture

- User Interface
- Knowledge Management
- Knowledge Search
- Reporting
- Data Management
- Knowledge Repositories
Support for Architecture Design

- Support for case-based reasoning by human expert
- Repository of reusable architectural artifacts
- Capture/access rationale for design decisions
- Catalogue of architecture and design patterns/tactics
- Search architectural artifacts and knowledge
Capturing and Reusing Cases of Design Decisions

Design Decision Captured as a Case
Searching Design Decision Cases

Using a Design Decision Case
Navigating the Knowledge Base

Template for Capturing and Representing Patterns
Support for Architecture Documentation

- Templates for documenting design decisions
- Store architectural models and documents
- Support for standards such as IEEE 1471-2000
- Represent architectural decisions using views
- Attach process knowledge to architectural artifacts

Structuring and Representing Architecture Decisions
Modifying an Architecture Decisions

Modifying a Pattern
Support for Architecture Evaluation

- Repository of general scenario to support QAWs
- Capture rationale and contextual information surrounding design decisions
- Search and view rationale for previous design decisions
- Documenting findings of evaluating design decisions
- Categorize findings in suitable risk themes
- Generate evaluation reports for the management

---

**General Scenario Captured in BRedB**

<table>
<thead>
<tr>
<th>General Scenario Listing</th>
<th>Model</th>
<th>Name</th>
<th>Source</th>
<th>Date</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed General Scenarios</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BR1.1</td>
<td>User Defined</td>
<td>Tue, Dec 19 2006 10:03</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BR1.2</td>
<td>User Defined</td>
<td>Tue, Dec 19 2006 10:03</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BR1.3</td>
<td>User Defined</td>
<td>Tue, Dec 19 2006 10:03</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BR1.4</td>
<td>User Defined</td>
<td>Tue, Dec 19 2006 10:03</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Accepted General Scenarios</th>
<th>Model</th>
<th>Name</th>
<th>Source</th>
<th>Date</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>N/A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rejected General Scenarios</th>
<th>Model</th>
<th>Name</th>
<th>Source</th>
<th>Date</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>N/A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Utility Tree of Concrete Scenarios

The Utility Tree is a visualization tool used to manage and prioritize concrete scenarios. It helps in identifying the critical aspects of a scenario and assigns weights to each parameter.

### Parameters
- **Performance**:
  - **Response Time**: 1 second
  - **Internal message transfer**: 1 second [High]
  - **Reduction based on comparison**: 1 second [High]

- **Reliability**
  - **Change in size**: 2 weeks
  - **New message**: 1 week
  - **New message**: 1 month
  - **Change from IPv6 to IPv4**: 1 percent/month [Medium]

- **Availability**
  - **Survive single network failure**: 1 week
  - **Random failure**: 1 second
  - **Utilization of Commanders and new Commander role in operation within 5 minutes**: [High]

### Viewing Details about a Concrete Scenario

The Viewing Details feature provides comprehensive information about a specific concrete scenario, including its name, description, quality factor, complexity level, importance, contact details, response time, and more.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name</strong></td>
<td>Name simulations with debug enabled</td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td>Name simulations with debug enabled</td>
</tr>
<tr>
<td><strong>Quality Factor</strong></td>
<td>Meet requirements</td>
</tr>
<tr>
<td><strong>Complexity Level</strong></td>
<td>Low (Default)</td>
</tr>
<tr>
<td><strong>Importance</strong></td>
<td>Low (Default)</td>
</tr>
<tr>
<td><strong>Source of Stimuli</strong></td>
<td>TUE Dec 2005 16:42</td>
</tr>
<tr>
<td><strong>Status</strong></td>
<td>Proposed</td>
</tr>
<tr>
<td><strong>Created By</strong></td>
<td>Master Administrator</td>
</tr>
<tr>
<td><strong>Classification</strong></td>
<td>Unclassified (Default)</td>
</tr>
</tbody>
</table>

### References
- **References**:
  - **Name**: Administrator
  - **Created By**: Administrator
  - **Tasks**: Done
  - **Findings**: No significant findings

### Controls
- **View**: Details about a concrete scenario.
- **Edit**: Modify the scenario details.
- **Cancel**: Exit without changes.
Documenting and Viewing Findings

Support for Architecture-Centric Development
BRedB Case Study

- Australian Defense Science Technology Organization (DSTO)
  - Avionics architecture assessment
  - Long lived, high cost projects
- BRedB used for avionic architecture evaluation
  - Quality attributes
  - Evaluation framework
  - Risk management

BRedB Supported Architecture Evaluation Process

- Requirements:
  - Functional
  - Verification
- Specification Define
- Problem Description
- Quality Define
- Desired Quality Attribute Measures
- Analytical Model / Reasoning Framework
- Quality Attribute Measures and Risks
- Architecture Design
- Architecture Description
- Design Tactics Patterns
- Scenarios
Initial Findings

• BRedB added considerable rigour to the DSTO evaluation process
  – Repository of expert knowledge in general scenarios
  – Archive of past project experience valuable over time
  – effective mechanism to organise and query large amounts of architecture knowledge

• BRedB should be useful if you:
  – are outsourcing/off-shoring/purchasing your systems?
  – have superhero architects prone to sudden departure or large pay increase demands

• More studies and R&D needed
  – We’re keen to work with interested partners ...

Conclusion and Future work

• Software architecting is receiving significant attention – several methodologies proposed and promoted but little attention on managing architectural knowledge

• Limited reusability of architectural artefacts.

• We have identified/developed techniques to provide knowledge management support for architecting

• PAKME – Supports the knowledge management tasks

• Future work includes:
  – Contextual search for retrieving task-related information
  – Converting PAKME into Eclipse plug-in
  – Assessing the value of PAKME through more industrial trials
Thank You