A Playbook for Early Architecture Analysis

John Klein (jklein@sei.cmu.edu)
Phil Bianco
Rick Kazman
James Ivers

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213
An architecture is not inherently good or bad – It is *fit for purpose*.

A good architecture is one that allows a system to meet its functional, quality attribute, and lifecycle requirements.

- Functionally correct
  - Throughput
  - Latency
  - Availability
  - …

- Evolvable
  - Composable
  - Integrable
  - …

Observable in the running system
- Early: Model and simulate
- Mid: Continuous integration
- Later: Final implementation

Observable in development artifacts
- Gather data and measurements
- Use analysis methods
- Make judgments
Architecture Analysis Challenge

Challenge: Make judgments about architectural qualities such as integrability…
• More systematic and repeatable
• Less reliant on the expertise of the analyst

Solution: Develop a playbook for the analysis process for each quality
• Applicable throughout the lifecycle, from early concept through implemented system
• Include practical information about specifying requirements, design mechanisms, measurements, and evaluation checklists and questionnaires

Outcome: Acquirers can verify that they are getting an architecture that will allow the system to meet its functional, quality attribute, and lifecycle requirements
## Architecture Analysis Playbook

<table>
<thead>
<tr>
<th>Phase</th>
<th>Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preparation</td>
<td>Step 1–Collect artifacts</td>
</tr>
<tr>
<td>Orientation</td>
<td>Step 2–Identify the mechanisms used to satisfy the requirement</td>
</tr>
<tr>
<td></td>
<td>Step 3–Locate the mechanisms in the architecture</td>
</tr>
<tr>
<td></td>
<td>Step 4–Identify derived decisions and special cases</td>
</tr>
<tr>
<td>Evaluation</td>
<td>Step 5–Assess requirement satisfaction</td>
</tr>
<tr>
<td></td>
<td>Step 6–Assess impact on other quality attribute requirements</td>
</tr>
<tr>
<td></td>
<td>Step 7–Assess the cost/benefit of the architecture approach</td>
</tr>
</tbody>
</table>

The quality of the architecture artifacts—breadth, depth, and completeness—sets the degree of confidence in the results:

- **Early:** Lower confidence may be acceptable → simpler analysis in Step 5 using lower quality artifacts
- **Later:** Higher confidence needed → deeper and more precise analysis using higher quality artifacts
Phase 1 – Preparation

Step 1 – Collect artifacts:
• Requirement to validate – playbook guidance includes quality attribute general scenario to specify a measurable requirement
• Other requirements – architecture design is a tradeoff process, improving one quality can reduce another
• Architecture documentation, for example:
  - Early – list of architecture approaches
  - Later – complete architecture views

Does the quality of the artifacts support the needed degree of confidence in the analysis results?
Phase 2 – Orientation

Step 2 – Identify the mechanisms used to satisfy the requirement
- Mechanisms: Patterns, tactics, frameworks
- Look at rationale discussions, trade studies, architecture views

Step 3–Locate the mechanisms in the architecture
- How is the mechanism used?
- For example, the trade study says that the architecture uses a “pipe and filter approach for the signal processing pipeline”
  - How many stages in the pipeline? Fixed or able to vary?
  - Playbook guidance includes checklists and questionnaires that identify these questions.

Step 4 – Identify derived decisions and special cases
- Derived decision: If the pipeline can vary, when is that defined – code or a configuration file?
- Special case: If configuration can change while the system is running, what happens to data in the pipeline?

What are the questions that the analysis must answer?
Phase 3 – Evaluation

Step 5 – Assess requirement satisfaction

- Answer the questions noted in Phase 2: Are the chosen mechanisms sufficient? Are derived decisions sufficient? Are any special cases addressed?
- Playbook guidance includes measurements, checklists, and questionnaires to support this step
- Identify issues: Possible gaps, deficiencies, ambiguities

Step 6 – Assess impact on other quality attribute requirements

- “Everything is a tradeoff”
- How do the mechanisms and decisions assessed in Step 5 affect other qualities?
- For example, a pipe and filter mechanism can increase latency and configuration files can impact testability.

Step 7 – Assess the cost/benefit of the architecture approach

- This is a judgement call – is any added complexity needed?

Is the proposed approach sufficient and necessary?
Where to find all the details…

Series of technical reports, one per quality attribute

- **Integrability** is released, **Maintainability** and **Robustness** are coming soon

Use to construct requirements and evaluate architecture against requirements

Each report contains

- Definitions
- Measures
- Scenario templates and examples
- Mechanisms – patterns and tactics
- Analysis methods – questionnaires, checklists, measurement
- Evaluation playbook

http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=637375

jklein@sei.cmu.edu

info@sei.cmu.edu