Evaluating Software Architectures for Real-Time Systems

R. Kazman, M. Klein, P. Clements
Software Engineering Institute
Carnegie Mellon University

Methodology for evaluating software architectures of real-time systems

Uses the Architecture Tradeoff Analysis Method (ATAM)

- ⌘Articulates Quality Goals
- ⌘Identifies Sensitivity Points
- ⌘Identifies Tradeoff Points

Scenarios are used to elicit quality goals

- ⌘Use Cases
- ⌘Growth Scenarios
- ⌘Exploratory Scenarios

Attribute Taxonomies are collections of knowledge from expert communities about quality attributes such as performance, reliability, security, or modifiability

Attribute information in the taxonomies is placed in one of three categories

- External stimuli
- Architectural parameters
- Responses

Taxonomies help ensure attribute coverage and offer a rationale for asking elicitation questions

Example Taxonomy

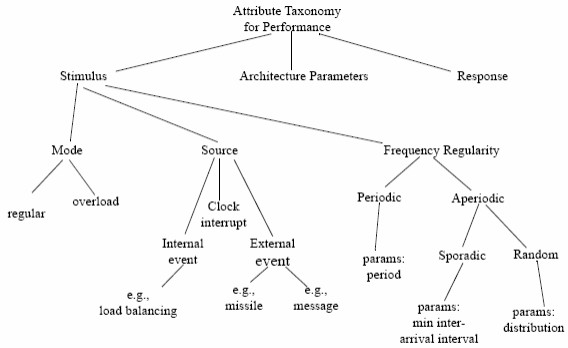

Figure 1: Performance Taxonomy: Stimuli

As a scenario is traced through the architecture, the elicitation questions uncover attribute specific information

Screening questions are also asked to guide or focus the elicitation

Outcome of elicitation is the identification of

- Sensitivity points
- Tradeoff points

Once these points have been identified, Attribute-based architecture styles (ABAS) are applied

- Routine and predictable design
- Standard set of analysis questions
- tight link between design and analysis

An ABAS has four parts

- Problem description
- Stimuli/Responses
- Architectural Style
- Analysis

ABAS Example

Part 1: Problem Description

Consider a single processor on which multiple processes reside, each of which perform computations on their own input data stream. Each final output from the system must be produced within a specified time interval after the arrival of an input, after all computations have been performed. We refer to the input data as a *message*. The requirement then is to completely process each message with a specified bounded end-to-end latency.

Part 2: Stimulus/Response

Stimuli: two or more periodic or sporadic input streams

Response: end-to-end worst-case latency

ABAS Example

Part 3: Architectural Style

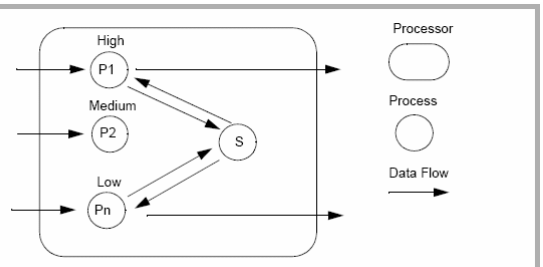

Figure 4: Synchronization ABAS

Part 4: Analysis

Latency for a process:

$$l_{n+1} = \sum_{j=1}^{i-1} \left\lceil \frac{l_n}{T_j} \right\rceil C_j + C_i + B_i$$

Qualitative analysis heurisitics are coarse grained analyses of the ABAS

Questions capture the essence of problems discovered by more formal methods

Example qualitative analysis heuristics

- If this architecture has real-time performance requirements and consists of multiple processes that share a resource then what process prioritization strategy is used including the priority used during the critical sections?
- If this architecture includes layers/facades, are there any places where the layers/facades are circumvented?

Analysis

ATAM is the application of a common engineering practice - the critical design review - to software systems

Analysis - Key Points

Identifies critical system attributes that are most sensitive to design decisions

Uses experiential evidence to evaluate the system architecture with respect to the critical attributes

Analysis - Advantages

Is very easy and relatively inexpensive

Catches critical design flaws early

Analysis - Disadvantages

Is somewhat limited - only addresses architecture, does not address whether the system is functionally correct

Is very dependent on good architectural representation - suggests 4+1 views, would presumably work with UML