

Architecture and Architecture-Centric Process

CSCI 475

Fall 2007

Based on Philippe Kruchten's "The Rational Unified Process An Introduction"

Models and RUP

- A *model* is a simplification of reality that helps us master a large, complex system that cannot be comprehended easily in its entirety.
- The model is not the reality, but the best models are the ones that stick very close to reality.
- A large part of RUP focuses on modeling.
- Multiple models are needed to address different concerns

Architecture

- Software architecture is commonly defined in terms of components and connectors (<http://www.bredemeyer.com/whatis.htm>)
- Components are identified and assigned responsibilities that client components interact with through "contracted" interfaces.
- Component interconnections specify communication and control mechanisms, and support all component interactions needed to accomplish system behavior.

Alternative View -- Architecture

- Describe a system so designers, programmers, users, and managers (*stake-holders*) would be able to
 - Understand what the system does
 - Understand how the system works
 - Work on one piece of the system
 - Extend the system
 - Reuse part of the system to build another one

Alternative View -- Architecture

- Architecture is what remains when you cannot take away any more things and still understand the system and explain how it works.

Architecture Today

- For an organization to adopt an architecture focus, three things are required
 - An understanding of the purpose
 - An architectural representation
 - An architecture process

Users of an Architecture

- The system analyst, who uses it to organize and articulate the requirements and to understand the technological constraints and risks
- End users or customers, who use it to visualize at a high level what they are buying
- The software project manager, who uses it to organize the team and plan the development
- The designers, who use it to understand the underlying principles and locate the boundaries of their own designs
- Other development organizations (if the system is open), which use it to understand how to interact with it
- Subcontractors, who use it to understand the boundaries of their chunk of development
- Architects, who use it to reason about the evolution or reuses.

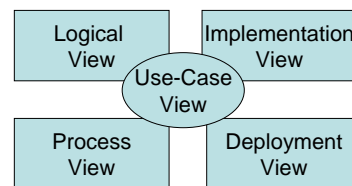
Multiple Views of an Architecture

- Similar to blueprints of a building
 - Floor plans
 - Elevations
 - Electrical cabling
 - Water pipes, central air-condition, and ventilation
 - The look of the building in its environment (in sketches)

Views in a Software Architecture

- The 4+1 View Model
 - The logical view
 - The implementation view
 - The process view
 - The deployment view
 - The user-case view

The 4+1 Views



The Logical View

- This view describes what the system does and addresses the functional requirements of the system
 - In an airline reservation system example
 - Flight database
 - Agent component
 - End-user component
 - Ticketing component
 - Billing component

The Implementation View

- This view describes the organization of static software modules and addresses the issue of ease of development, management, reuse and others.
 - In an airline reservation system example
 - Java package for database handling
 - Java classes for graphical user interface with end users and agents
 - Java library for scheduling (finding optimal route for a given request)

The Process View

- This view describes the concurrent aspects of the system at run-time and addresses issues such as fault tolerance, deadlock, response time and throughput.
 - In an airline reservation system example
 - Which java module should start first
 - How multiple agents can access a shared database

The Deployment View

- This view describes how the various executables and other run-time libraries are mapped to the underlying platforms and addresses issues such as installation, performance, and system requirements
 - In an airline reservation system example
 - The agent reservation component is installed on every workstation
 - The database is a centralized one

The Use-Case View

- Contains a few key scenarios or use cases
- Initially these use cases are used to drive the discovery and design of the architecture
- Later these use cases are used to validate the different views
 - In an airline reservation system example
 - Customer books a ticket
 - Customer makes change to the original ticket

Architecturally Significant Elements

- Major classes, in particular the classes that model major business entities
- Architectural mechanisms that give behavior to these classes
- Patterns and frameworks
- Layers and subsystems
- Interfaces
- Major processes, or threads of control

Architecture Is More Than Blueprint

- Primary objectives of the elaboration phase
 - Building the architecture
 - Validating it
 - Baselineing it (prototype)
- Most important artifact associated with the architecture is an architectural prototype
 - A prototype that implements the most important design decisions
 - Validate the decisions (test and measure them)

An Architecture-Centric Process

- RUP defines two primary artifacts related to architecture
 - The software architecture description (SAD)
 - The architectural prototype
- Three other artifacts rooted from the above two
 - Design guidelines
 - The product structure
 - The team structure

Software Architect

- Responsible for the architecture in elaboration phase
- Other team members are also involved
 - Designers focus on architecturally significant classes
 - Integrators integrate major software components
 - Testers test the architectural prototype for performance and robustness

Other Architectural Concepts

- Architectural style – may be defined by an architectural framework, set of patterns
 - Examples: pipe-and-filter, client-server, event-driven
- Architectural mechanism – a pattern that provides a common solution to a common problem
 - Examples: DBMS, an event broadcasting system, a transaction server