



Software Design and Architecture

Architectural Design

Sajid Anwer

Department of Software Engineering,
FAST-NUCES, CFD Campus



Lecture Outline

- Impact of System Size on Architectural Design
- Fundamentals of Architectural Design
- Architectural Design Elements



Lecture Material

- Software Architecture, Foundation, Theory, and Practice (Ch#3)

Impact of Software Systems' on Architectural Design

- Before we define software architecture, we need to define systems in *general* and *software systems* in particular
 - » The system itself can be a part of a *larger system* as a subsystem
 - » A system is a collection of *parts* and a *set of principles how those parts are connected* in a whole
 - » A system is a closed entity that serve a *single purpose*
 - » Performs a certain function or set of functions
 - » Has *input* and produces *output* according to its function(s)

Impact of Software Systems' on Architectural Design

- All software systems depend largely on hardware, networking infrastructure, etc.
- Web browser
 - » Parts: Displaying engine (HTML parser, GUI, CSS formatter)
 - » Networking module: HTTP client, XMLHttpRequest module
 - » Data manager: Bookmarks, Cookie Manager



Fundamentals of Architectural Design

- Software architecture designed based on,
 - » System requirements (functional, quality and constraints)
 - » Contextual information (four context facets)
 - » Other user stories and narratives (informal)



Fundamentals of Architectural Design

- Software architecture describes:
 - » a software system as a whole
 - » breaks it down into parts
 - » specifies how parts are put together (e.g. how they communicate) to meet a needed purpose.
- Architectural design is the design of the overall *structure* and *behaviour* of the system

Prescriptive vs. Descriptive Architecture

- prescriptive architecture
 - » It is the *as-conceived* or *as-intended* architecture
- descriptive architecture
 - » It is the *as-implemented* or *as-realized* architecture
- When a system *evolves*, ideally its prescriptive architecture is modified first
- In practice, the system – and thus its descriptive architecture – is often directly modified
- This happens because of
 - » Developer sloppiness
 - » Perception of short deadlines which prevent thinking through and documenting
 - » Lack of documented prescriptive architecture
 - » Need or desire for code optimizations

Architectural Degradation

- **Architectural drift** is introduction of *design decisions* into a system's descriptive architecture that
 - » are not included in, encompassed by, or implied by the prescriptive architecture
 - » but which *do not violate* any of the prescriptive architecture's design decisions
- **Architectural erosion** is the introduction of architectural design decisions into a system's descriptive architecture that violate its prescriptive architecture.
- **Architectural recovery** is the process of *determining* a software system's architecture from its *implementation-level artifacts*



Architectural Design

- A *software system's architecture* should be a composition and interplay of different elements/components
 - » Processing
 - » Data, also referred as information or state
 - » Interaction
- The major *architectural components* of any system are the *software* and the *hardware*.

Architectural Design

- The major *software components* of the system being developed have to be identified and then allocated to the various hardware components on which the system will operate.
 - » Data storage
 - File, structured database, video, audio
 - » Data access logic
 - SQL
 - » Application logic
 - Programs, functional requirements realization
 - » Presentation logic
 - Web pages, reports
- The major *hardware components* of the system are;
 - » Client computers
 - » Servers
 - » Network

Architectural Design -- Components

- Elements that encapsulate processing and data in a system's architecture are referred to as *software components*.
- **Definition**
 - » A *software component* is an architectural entity that
 - encapsulates a subset of the system's functionality and/or data
 - restricts access to that subset via an explicitly defined interface
 - has explicitly defined dependencies on its required execution context
- Components typically provide *application-specific services*

Architectural Design -- Connectors

- In complex systems, *interaction* may become more important and challenging than the functionality of the individual components
 - » A **software connector** is an architectural building block tasked with effecting and regulating interactions among components
- In many software systems connectors are usually simple procedure calls or shared data accesses
 - » Much more sophisticated and complex connectors are possible!
- Connectors typically provide application-independent interaction facilities

Architectural Design -- Connectors

- Procedure call connectors
 - » Directly implemented in programming languages.
 - » Typically enable *synchronous* exchange of data and control between pairs of components.
- Shared data access connectors
 - » In form of nonlocal variables or shared memory.
 - » Allow multiple software components to interact by reading from and writing to the *shared facilities*.

Architectural Design – Examples of Connectors

- Distribution connectors

- » Typically encapsulate network library application programming interfaces (APIs) to enable components in a distributed environment to interact.
- » Usually coupled with more basic connector to insulate the interacting components from system distribution details.
- » For example, Remote Procedure Call (RPC) connects.

- Adaptor connectors

- » Preexisting components, which may not have been tailor-made for the given system.
- » Wrappers and glue code are two common kinds of adaptor connectors.

Architectural Design – Configuration

- Components and connectors are *composed* in a *specific way* in a given system's architecture to accomplish that system's objective
 - » An *architectural configuration*, or topology, is a set of specific associations between the components and connectors of a software system's architecture

Architectural Design – Configuration

