

COMP3027J

Software Architecture

Introduction

DENG, YONGJIAN

Faculty of Computer Science, BJUT

Data Mining & Security Lab (DMS Lab)



Outline

1. History of Software Development
2. Definition of Software Architecture
3. Related Concepts of Software Architecture
4. Influencing Factors of Software Architecture
5. Values of Software Architecture



Outline

1. History of Software Development
2. Definition of Software Architecture
3. Related Concepts of Software Architecture
4. Influencing Factors of Software Architecture
5. Values of Software Architecture



Review of Software Development

The size and the complexity of software is becoming
larger and more complex

The application areas of software: science computing, manufacturing, commerce, education and amusement

The abstraction level of software is becoming ***higher***

- † Machine Language → Assemble Language → Advanced Language → Application Framework
- † Structure-Oriented Programming → Object-Oriented Programming → Aspect-Oriented Programming



Outcome of Software Development

Good architecture design has always been a major factor in determining the success of a software system.

The architecture and designing is more important than the data structure and the program algorithm.



Outline

1. History of Software Development
2. Definition of Software Architecture
3. Related Concepts of Software Architecture
4. Influencing Factors of Software Architecture
5. Values of Software Architecture



Architecting a Kennel

Can be built by one person

- † Minimal modeling
- † Simple process
- † Simple tools



Architecting a House

Built most efficiently and
by a team

- † Modeling
- † Well-defined process
- † Power tools



Architecting a High Rise



Differences

- † Scale
- † Process
- † Cost
- † Schedule
- † Skills and development teams
- † Materials and technologies
- † Stakeholders
- † Risks



Philosophy



Edsger Dijkstra

“…the larger the project, the more essential the structuring!”(1968)



Definition of Software Architecture

Definition

The software architecture of a program or computing system is the ***structure*** or structures of the system, which comprise ***software elements***, the externally ***visible properties*** of those elements, and the ***relationships*** among them. — — — 《Software Architecture in Practice》, Addison-Wesley 1997

Definition

Architecture is the organizational ***structure*** of a system. An architecture can be recursively decomposed into ***parts*** that interact through ***interfaces***, ***relationships*** that connect parts, and ***constraints for assembling parts***. Parts that interact through interfaces include classes, components and subsystems. — — — UML 1.3



Definition of Software Architecture

Definition

Software architecture is the ***fundamental organization*** of a system, embodied in its ***components***, their ***relationships*** to each other and the environment, and the ***principles*** governing its design and evolution.

— — — IEEE 1471-2000



Definition of Software Architecture

Software Architecture

- † Software Elements: functions, Interfaces, programs, class modules, layers, subsystems, clients/servers, etc.
- † Visible Properties: provided services, performance characteristics, fault handling, shared resource usage, etc.
- † Relations: composition mechanism and style of these elements.

An architecture is the result of a set of business and technical decisions.



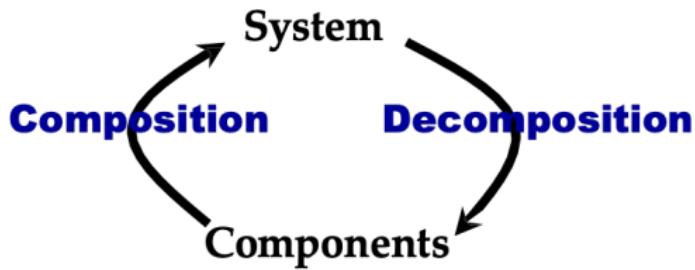
Definition of Software Architecture

A Software Architecture Include:

- † The constituent elements – Component
- † The interaction rules/mechanism – Connector
- † Constraint

In

- † The components comprised in the system, and the relationships or interaction mechanisms of those components.
- † Software Architecture Design = Decomposition + Composition



Definition of Software Architecture

Decomposition/Composition

- † Reducing the complexity of software design and construction
- † Controlling the risks of software development
- † Improving the efficiency of organization and management

Must be taken into account:

- † How do we break the system down into pieces?
- † Do we have all the necessary pieces?
- † Do the pieces fit together?

[Hundreds of definitions on CMU web page.](#)



Outline

1. History of Software Development
2. Definition of Software Architecture
3. Related Concepts of Software Architecture
4. Influencing Factors of Software Architecture
5. Values of Software Architecture



Related Concepts of Software Architecture

Component is a logical and functional unit of the system.

- † A component may be divided into more little unit of components.
- † A component serves certain responsibilities.
- † The component is an abstract and conceptual word, it will be different specific objects (eg., modules, subsystems, layers, packages, classes, etc.) in different scenarios.



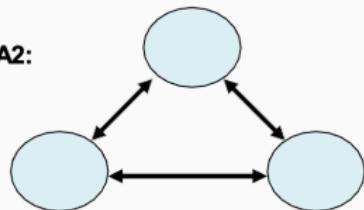
Related Concepts of Software Architecture

Connector represents interaction rules or mechanisms among components.

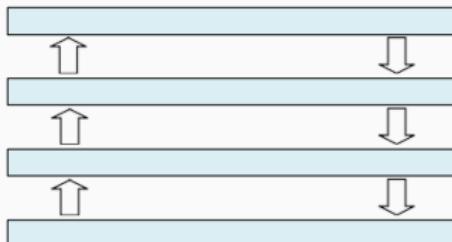
SA1:



SA2:



SA3:



Related Concepts of Software Architecture

Definition

Functional Property of the Software Architecture: the characters of the SA that meets the functional requirements.

Definition

Non-Functional Property of the Software Architecture: the characters of the SA that meets the non-functional requirements. For example, performance, portability, flexibility/extensibility, reliability/security.



Framework

- † A framework is a reusable application infrastructure for a specified problems.
- † Including necessary basic components for the specified problems.
- † Interaction mechanisms and constraints among components is comprised.
- † A context or environment for the applications developed based on the framework is provided.

Commonly, a framework mainly presents a class library. For example: .NET Framework, JavaEE Framework etc.



Summary of Software Architecture

- † Architecture is at a high-enough level of abstraction that the system can be viewed as a whole.
- † At the architectural level, all implementation details are hidden.
- † The architecture must support the functionality required of the system.
- † The architecture must conform to the system qualities (a.k.a. non-functional requirements) : performance, security and reliability, flexibility or extensibility.



Outline

1. History of Software Development
2. Definition of Software Architecture
3. Related Concepts of Software Architecture
4. Influencing Factors of Software Architecture
5. Values of Software Architecture



Influencing Factors of Software Architecture

- † Architectures are influenced by system stakeholders.
- † Architectures are influenced by the developing organization.
- † Architectures are influenced by the background and experience of the architects.
- † Architectures are influenced by the technical environment.



Architectures are Influenced by System Stakeholders

Architectures is many things to many different interested parties

- † End-user
- † Project manager
- † Customer
- † Architect
- † Maintainer
- † Other developers
- † System engineer
- † Developer

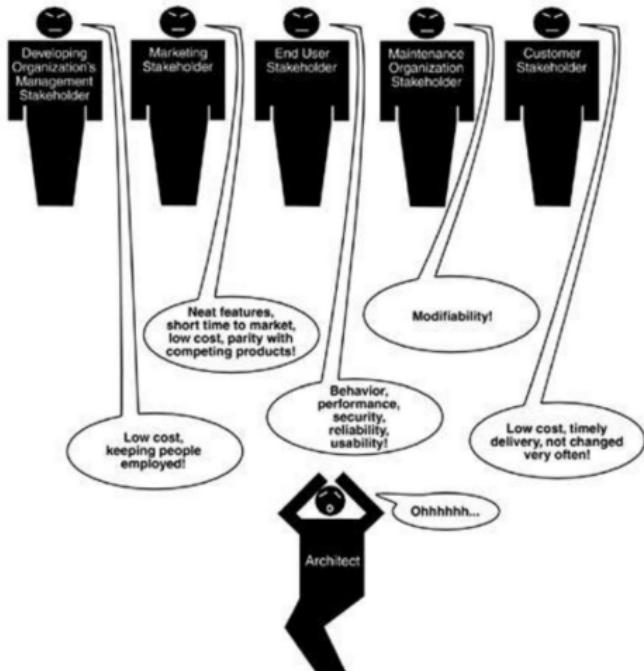
Multiple stakeholders

- † End-user Multiple views, multiple blueprints



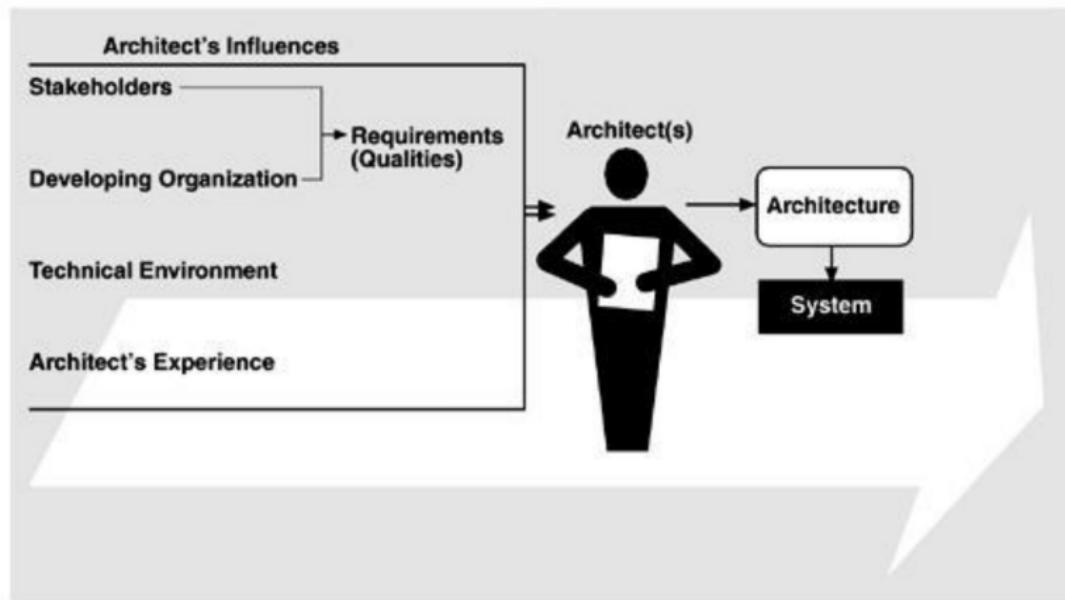
Architectures are Influenced by System Stakeholders

Influence of stakeholders on the architect



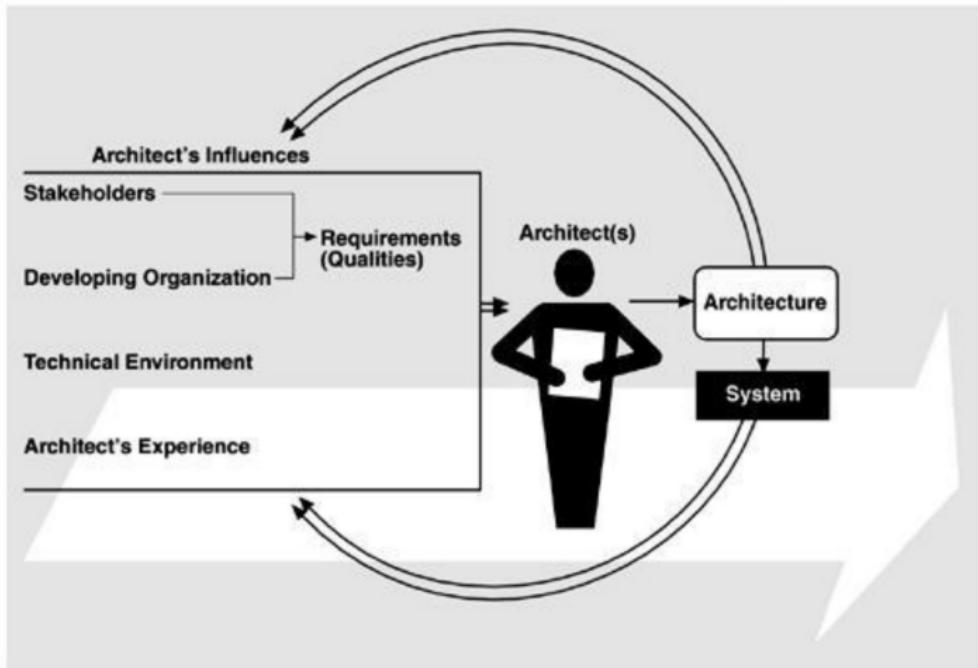
Architectures are Influenced by System Stakeholders

Influences on the architecture



The Feedback Loops

The Architecture Business Cycle

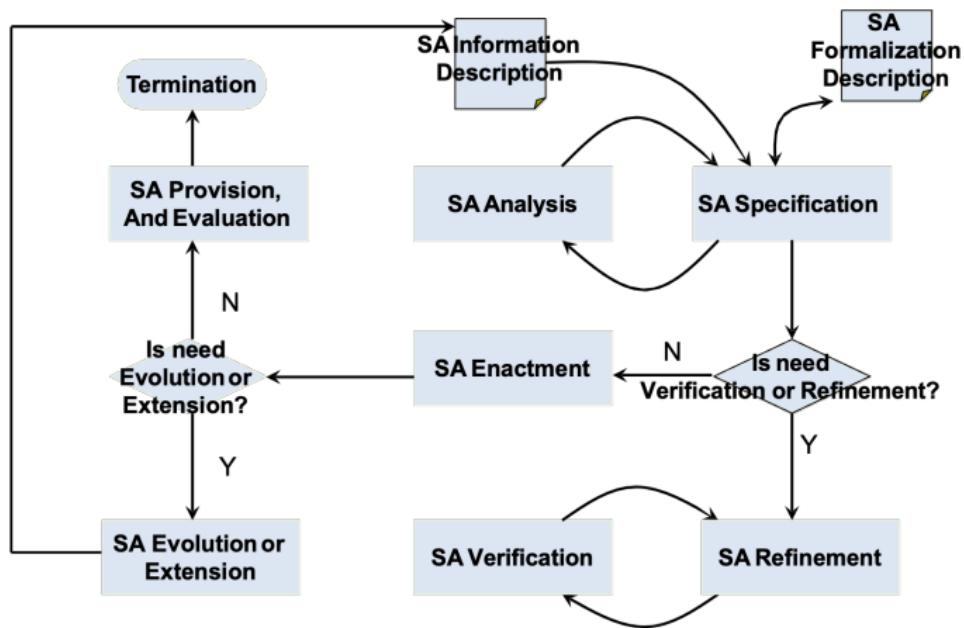


Process of Architecture Design

- † Creating the business case for the system.
- † Understanding the requirements.
- † Creating or selecting the architecture.
- † Documenting and communicating the architecture.
- † Analyzing or evaluating the architecture.
- † Implementing the system based on the architecture.
- † Ensuring that the implementation conforms to the architecture.



The Lifetime Model of Software Architecture



Why Is Software Architecture Important?

Architecture is the vehicle for stakeholder communication.

Architecture manifests the earliest set of design decisions.

- † The Architecture Defines Constraints on Implementation
- † The Architecture Dictates Organizational Structure
- † The Architecture Inhibits or Enables a System's Quality Attributes
- † Predicting System Qualities by Studying the Architecture
- † The Architecture Makes It Easier to Reason about and Manage Change
- † The Architecture Helps in Evolutionary Prototyping
- † The Architecture Enables More Accurate Cost and Schedule Estimates



Why Is Software Architecture Important?

Architecture as a transferable, re-usable model.

- † Software Product Lines Share a Common Architecture
- † Systems Can Be Built Using Large, Externally Developed Elements



Outline

1. History of Software Development
2. Definition of Software Architecture
3. Related Concepts of Software Architecture
4. Influencing Factors of Software Architecture
5. Values of Software Architecture



Values of Architecture

Architecture serves both ***technical*** and ***organizational*** purposes.

Organization side:

- † Communicate inside organization, and between customers and vendors
- † Provide the high-level information of systems
- † Costs and risks evaluating
- † Work allocation and project schedule



Values of Architecture

Architecture serves both ***technical*** and ***organizational*** purposes.

Technical side:

- † Meet system requirements and objectives
- † Specify the constraints of detailed design, construction and testing phrase
- † Enable flexible distribution/partitioning of the system
- † Reduce cost of maintenance and evolution
- † Increase reuse and integrate with legacy and third party software



Characteristics of a Good Architecture

- † Resilient
- † Simple
- † Approachable
- † Clear separation of concerns
- † Balanced distribution of responsibilities
- † Balances economic and technology constraints



Thank you!

