



# Software Architecture

Zhenyan Ji

[zhyji@bjtu.edu.cn](mailto:zhyji@bjtu.edu.cn)

Beijing Jiaotong University

Software Architecture

Zhenyan Ji, BJTU

1



## Contact information

**Zhenyan Ji**

**Associate Professor**

- Office Room: YF706
- Email: [zhyji@bjtu.edu.cn](mailto:zhyji@bjtu.edu.cn)
- Telephone: 13621070959

Software Architecture

Zhenyan Ji, BJTU

2

## Course Overview



- **Software Architecture**
- **The Principles for OO Design**
  - SRP—The Single-Responsibility Principle
  - OCP—The Open-Closed Principle
  - LSP—The Liskov Substitution Principle
  - DIP—The Dependency-Inversion Principle
  - ISP—The Interface-Segregation Principle
  - Six Principles of Package Design
- **Design Patterns**
- **Major Software Architecture Patterns**
- **Software Product Line**
- **Quality Attributes**

Software Architecture

Zhenyan Ji, BJTU

3

## Literature



- Software Architecture in Practice, 3rd Edition, By Len Bass, Paul Clements, Rick Kazman, Publisher: Addison Wesley, ISBN: 978-0321815736 。
- Agile Software Development, Principles, Patterns, and Practices, by Robert C. Martin, Publisher: Prentice Hall, ISBN-10: 0135974445, ISBN-13: 978-1292025940.
- The Design Patterns Java Companion, Free online book by James W. Cooper

Software Architecture

Zhenyan Ji, BJTU

4



## Literature

- 软件体系结构（第二版）覃征，邢剑宽，董金春，郑翔 编著 出版社：清华大学出版社，2008-3-1
- 软件体系结构（第2版）张友生 等编著 出版社：清华大学出版社，2006-11-1
- 软件体系结构理论与实践 马冲，江贺，冯静芳 编著 出版社：人民邮电出版社，2004-01
- 冀振燕编著，《UML 系统分析与设计教程(第二版)》（21世纪高等学校计算机规划教材精品系列），人民邮电出版社，ISBN 978-7-115-19795-5/TP，2009.5月。



## Useful Websites I

- Carnegie Mellon SEI: [http://www.sei.cmu.edu/ata/ata\\_init.html](http://www.sei.cmu.edu/ata/ata_init.html)
- Worldwide Institute of Software Architects: <http://www.wwisa.org/wwisamain/index.htm>
- Software Architecture: <http://www.softat.org/>
- 中国系统分析员: <http://www.csai.cn/>



## Useful Websites II

- OO Design Principles:
  - <http://c2.com/cgi/wiki?PrinciplesOfObjectOrientedDesign>
- Design Patterns:
  - [http://sourcemaking.com/design\\_patterns](http://sourcemaking.com/design_patterns)
  - [http://www.tutorialspoint.com/design\\_pattern/build\\_pattern.htm](http://www.tutorialspoint.com/design_pattern/build_pattern.htm)



## Evaluation Criteria

- Your final score consists of four parts.
  - Attendance Rate 5%
  - Active participation 5%
  - Assignments 40%
  - Written Exam 50%

# Any More Questions?



Software Architecture

Zhenyan Ji, BJTU

9

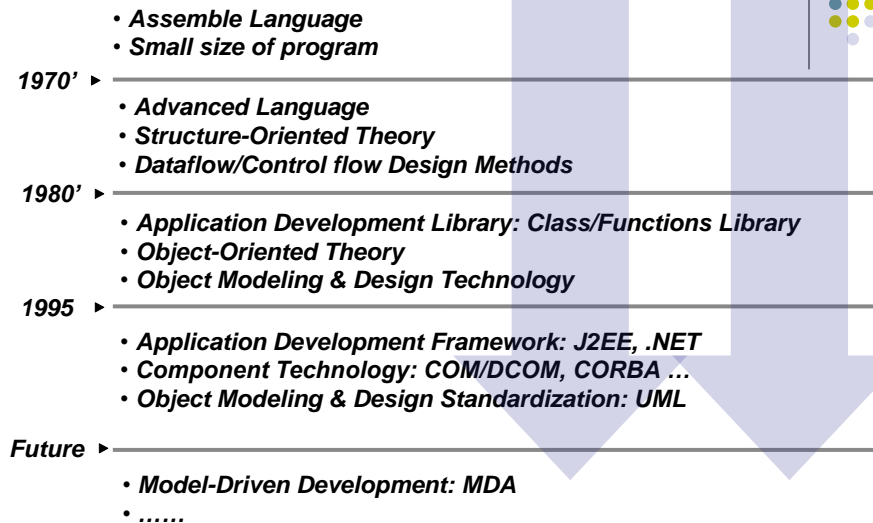
## Module 1: Software Architecture basics

### Content:

- History of Software Development
- Definition of Software Architecture
- Some Related Concepts of Software Architecture
- Where do architectures come from?
- Values of Software Architecture
- Current Research & Practicing in Software Architecture



## History of Software Development



Software Architecture

Zhenyan Ji, BJTU

11

## History of Software Development



- Review of the history of the software:
  - 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 *more high*.
    - Machine Language —> Assemble Language —> Advanced Language —> Application Framework
    - Structure-Oriented Programming —> Object-Oriented Programming —> Aspect-Oriented Programming

Software Architecture

Zhenyan Ji, BJTU

12

## History of Software Development



- Results of the development of software:
  - **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.**

Software Architecture

Zhenyan Ji, BJTU

13

## The Definition of Software Architecture



Can be built by one person  
Requires  
Minimal modeling  
Simple process  
Simple tools

Software Architecture

Zhenyan Ji, BJTU

14



## Architecting a house



Built most efficiently and timely by a team  
Requires  
Modeling  
Well-defined process  
Power tools

Software Architecture

Zhenyan Ji, BJTU

15



## Architecting a high rise



Software Architecture

Zhenyan Ji, BJTU

16





## Differences

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



## Definition of Software Architecture

- 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
- 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



- 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

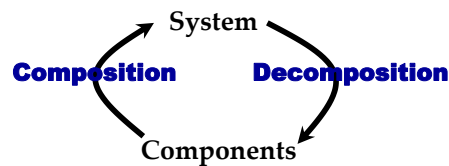


- Software Architecture
  - **Software Elements:** functions, Interfaces, programs, class modules, layers, subsystem, clients/servers etc.
  - **Visible Properties:** provided services, performance characteristics, fault handling, shared resource usage, and so on
  - **Relations:** composition mechanism 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**
- ***So, it can be defined briefly as:***
  - the components comprised in the system, and the relationships or interaction mechanisms of those components.
  - *Software Architecture Design = Decomposition + Composition*



Software Architecture

Zhenyan Ji, BJTU

21

# 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

But, we must consider

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

Software Architecture

Zhenyan Ji, BJTU

22

# Definition of Software Architecture



- Hundreds of definitions on CMU web page:  
<http://www.sei.cmu.edu/architecture/definitions.html>

## Related Concepts of SA



- **Component**
  - A logical and functional unit of the system.
- **Note:**
  - **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'll be different specific objects (for example, modules, subsystems, layers, packages, classes etc.) in different scenarios.**



## Related Concepts of SA

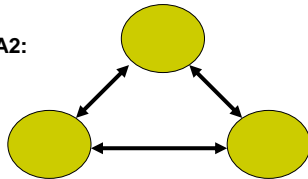
### • Connector

The interaction rules or mechanisms among components.

SA1:



SA2:



SA3:



Software Architecture

Zhenyan Ji, BJTU

25



## Related Concepts of SA

### • Functional Property of the SA

the characters of the SA that meets the functional requirements.

### • Non-functional Property of the SA

the characters of the SA that meets the non-functional requirements. For example,

- performance
- portability
- flexibility/extensibility
- reliability/security
- ...

Software Architecture

Zhenyan Ji, BJTU

26



## Related Concepts of SA

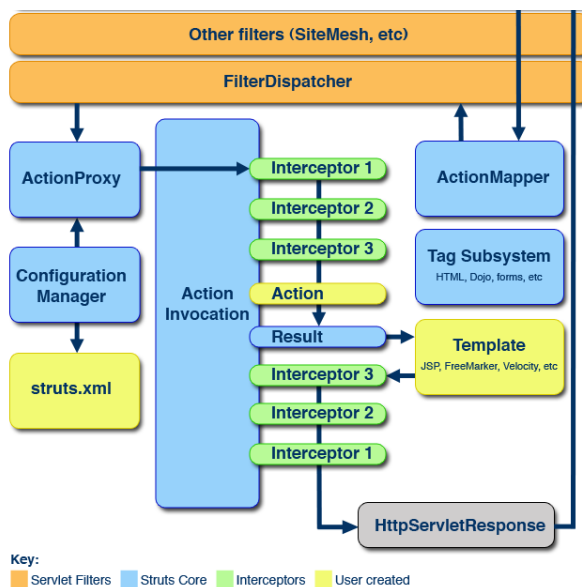
### • Framework

- A framework is a reusable application infrastructure for a specified problems.
- some necessary basic components for the specified problems
- interaction mechanism and constraints among components
- a context or environment for the applications developed based on the framework
- Commonly, a framework mainly presents a class library. For example: .NET Framework, JavaEE Framework etc.

Software Architecture

Zhenyan Ji, BJTU

27



## Architecture of Struts 2



Software Architecture

Zhenyan Ji, BJTU

28



## Comments for SA

- 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 (also known as non-functional requirements) : performance, security and reliability, flexibility or extensibility.

Software Architecture

Zhenyan Ji, BJTU

29



## Where Do Architectures Come From?

- **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.**

Software Architecture

Zhenyan Ji, BJTU

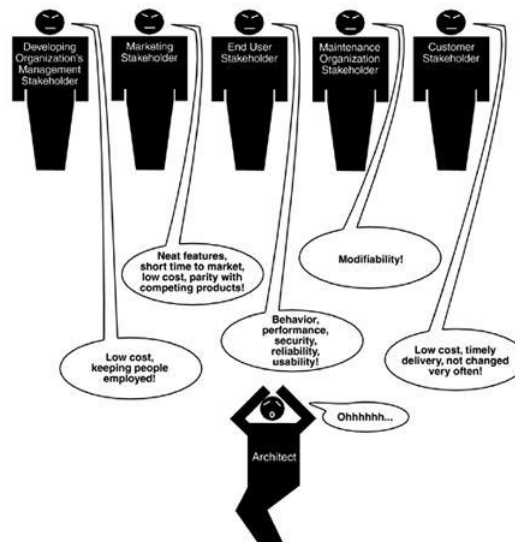
30

# Many stakeholders, many views



- Architecture is many things to many different interested parties
  - end-user
  - customer
  - project manager
  - system engineer
  - developer
  - architect
  - maintainer
  - other developers
- Multiple stakeholders
  - ➔ multiple views, multiple blueprints

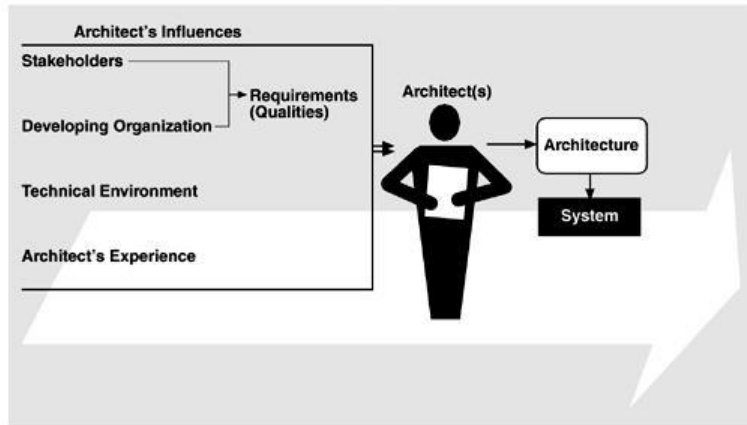
## Influence of stakeholders on the architect







## Influences on the architecture



Software Architecture

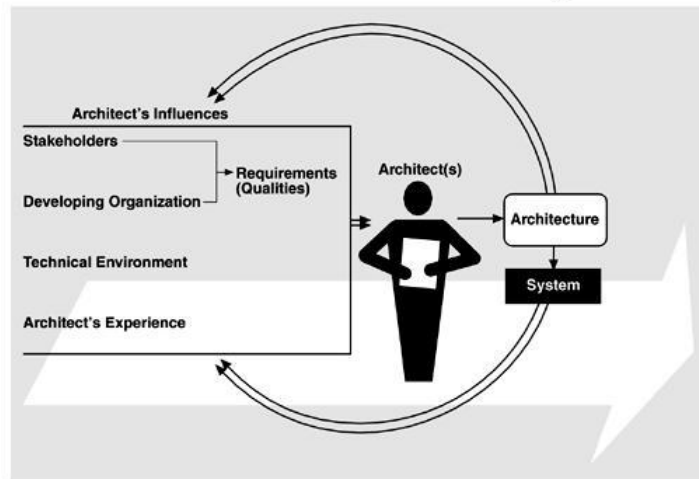
Zhenyan Ji, BJTU

33

## The feedback loops

- The architectures affect the factors that influence them.

### The Architecture Business Cycle



Software

34

# Architecture Business Cycle



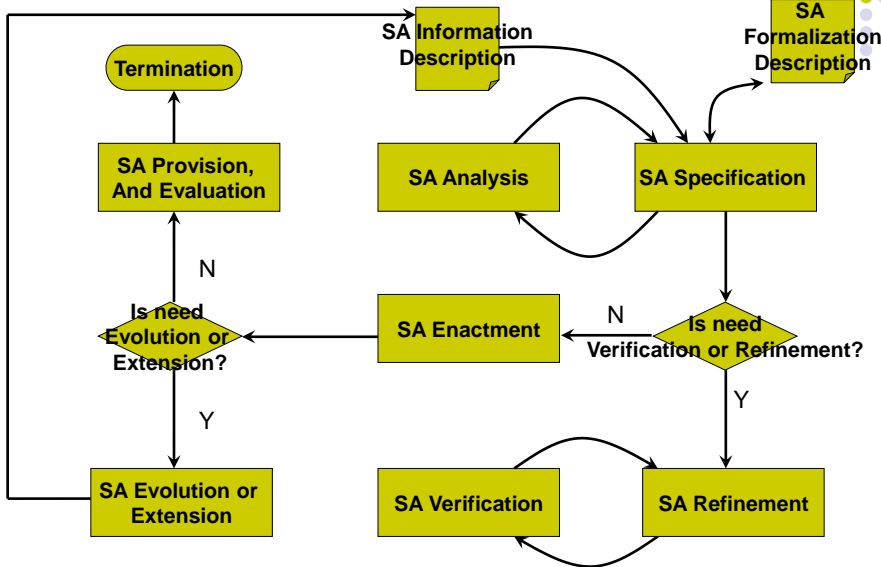
- Activities of architecture business cycle:
  - Creating the business case for the system
  - Understanding the requirements
  - Creating or selecting the architecture
  - Documenting and communicating the architecture
  - Analyzing or evaluating the architecture
  - Implementing the system based on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

Software Architecture

Zhenyan Ji, BJTU

35

## The lifetime model of SA



Software Architecture

Zhenyan Ji, BJTU

36

## 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*

Software Architecture

Zhenyan Ji, BJTU

37

## Why Is Software Architecture Important?



- *The Architecture Helps in Evolutionary Prototyping*
  - *The Architecture Enables More Accurate Cost and Schedule Estimates*
- **Architecture as a transferable, re-usable model.**
  - *Software Product Lines Share a Common Architecture*
  - *Systems Can Be Built Using Large, Externally Developed Elements*

Software Architecture

Zhenyan Ji, BJTU

38

## Values of Architecture



- Architecture serves both technical and organizational purposes:
- **Organization side:**
  - *communicating inside organization, and between customers and vendors*
  - *providing the high-level information of systems*
  - *Costs and risks evaluating*
  - *work allocation and project schedule*

## Values of Architecture



- **Technical side:**
  - *meet system requirements and objectives*
  - *Specify the constraints of detailed design, construction and testing phase*
  - *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

## Research of SA



- **Formalization Research**

**refers to** *how to describe the SA in specific rules*

*how to view and present the SA*

*ADL--- Architecture Description Language*

- **Verification & Evaluation Research**

**refers to** *how to verify and evaluate the SA whether it meets the constraints of functional and non-functional requirement.*

*ATAM---Architecture Tradeoff Analysis Method*



## Assignment1

- Select a research hotspot related with Software Architecture. Do extensive reading and write an overview of your survey.
- Tasks: Write a paper or report.