

Software Architecture

Session 3:
Architecture Driven Development

Objectives

- a) Recall the concept of Software Life Cycle
- b) Understand the concept of Software Architecture Development Cycle
- c) Understand the concept of Architecture Driven Development

Outline

1. Software Life Cycle
2. Software Architecture Development Cycle
3. Architecture Driven Development
4. Summary

Outline

1. Software Life Cycle
2. Software Architecture Development Cycle
3. Architecture Driven Development
4. Summary

What is a software life cycle?



Software Life Cycle

A software (development) life cycle, is **a structure imposed** on the development of a software product.

The **technical activities** of such a structure include:

- Analysis
- Design
- Implementation (a.k.a. construction, codification, development.)
- Testing
- Deployment (a.k.a. installation)
- Maintenance
- Retire

Software Development Life Cycle

Focuses on determining the requirements to meet for a new or altered product

Analysis

Focuses on defining the model for a system to satisfy specified requirements

Design

Implementation

Testing

Deployment

Maintenance

Part of the process where software engineers actually program the code for the system

Software Development Life Cycle

The modification of a software system after delivery to correct faults, to improve performance or other quality attributes.

Focuses on evaluating the system and determining that it meets its requirements.

Analysis

Design

Implementation

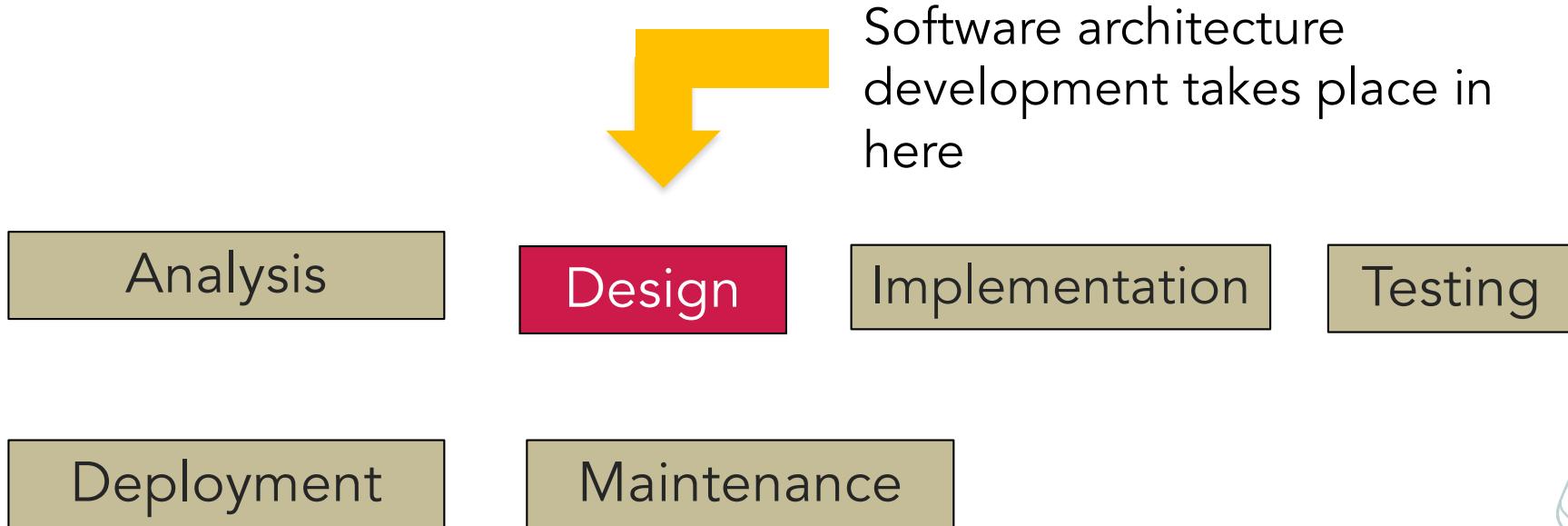
Testing

Deployment

Maintenance

Focuses on all of the activities that make a software system available for use.

Software Development Life Cycle



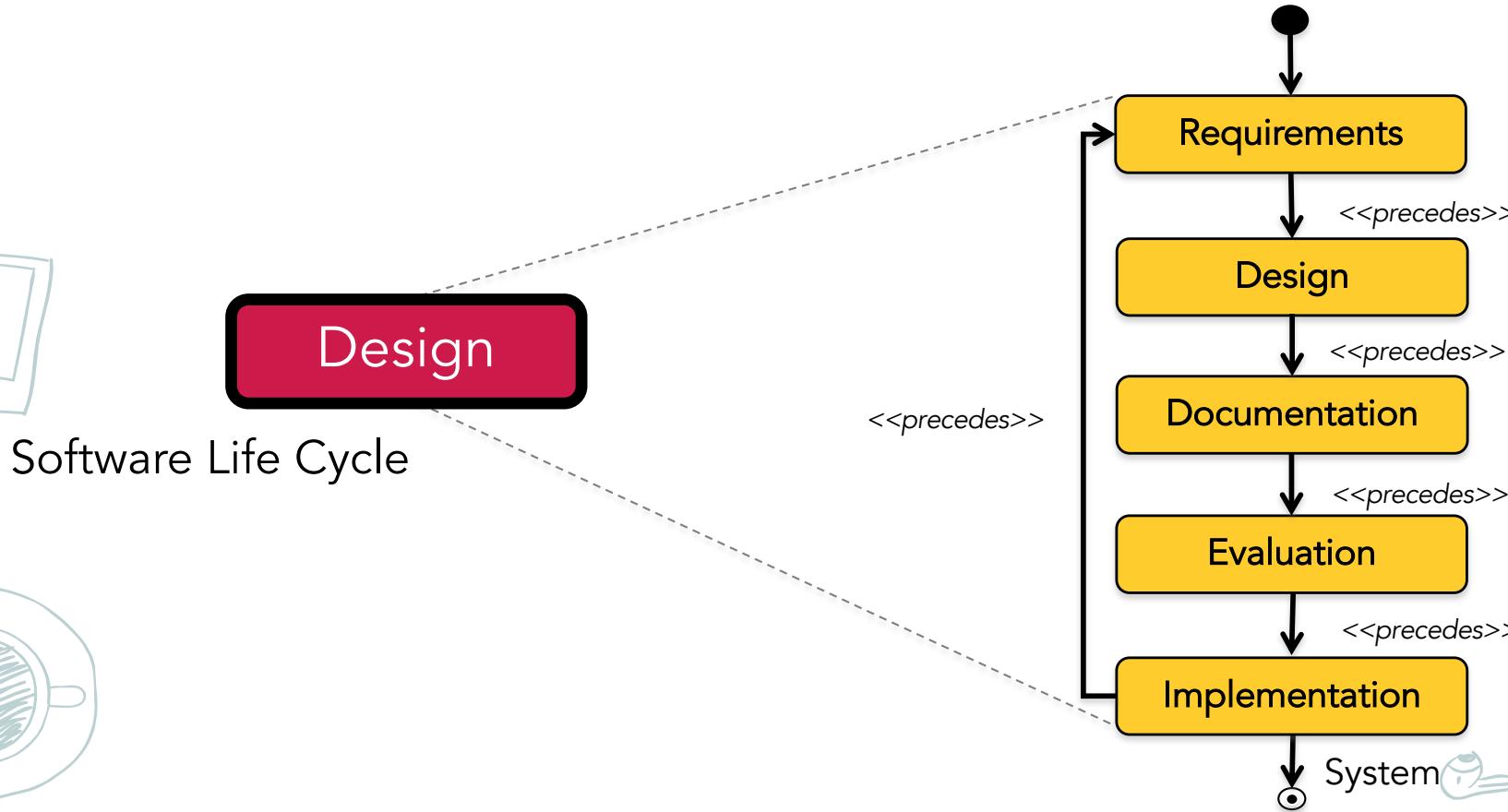
Outline

1. Software Life Cycle
2. Software Architecture Development Cycle
3. Architecture Centred Development
4. Summary

What is a Software Architecture Development Cycle ?



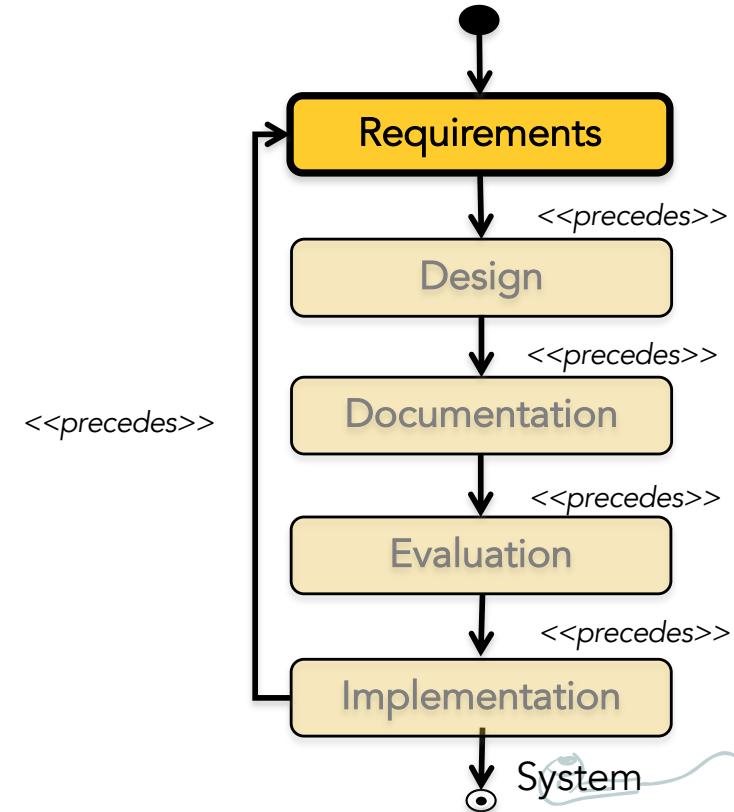
Software Architecture Development Cycle



Software Architecture Development Cycle

Identifying architectural requirements

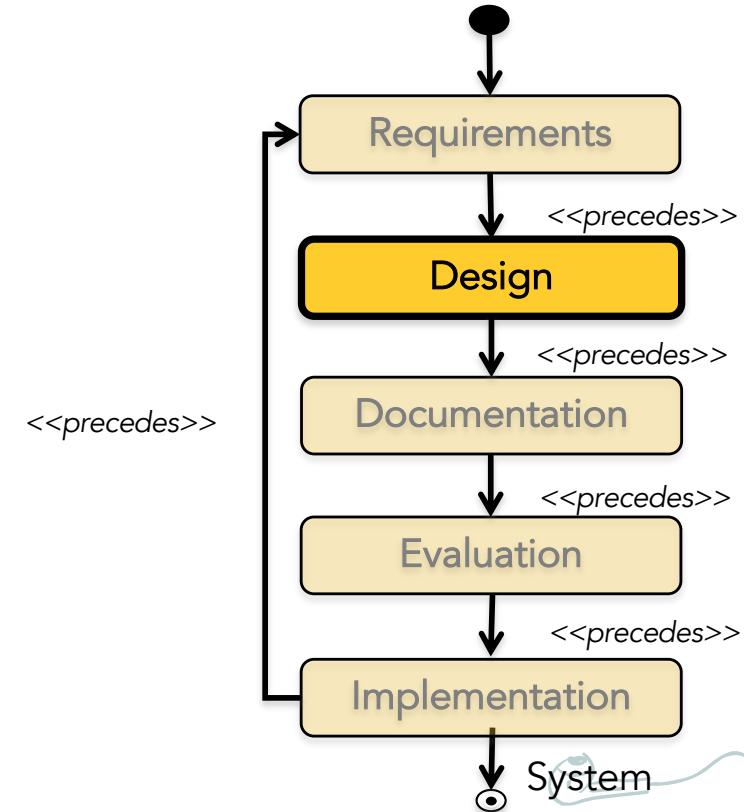
Among all the requirements **there are a few** that can be said to drive the design. These requirements will guide the architect to choose one set of architectural structures and components over another.



Software Architecture Development Cycle

Making design decisions,
creating architectural
structures

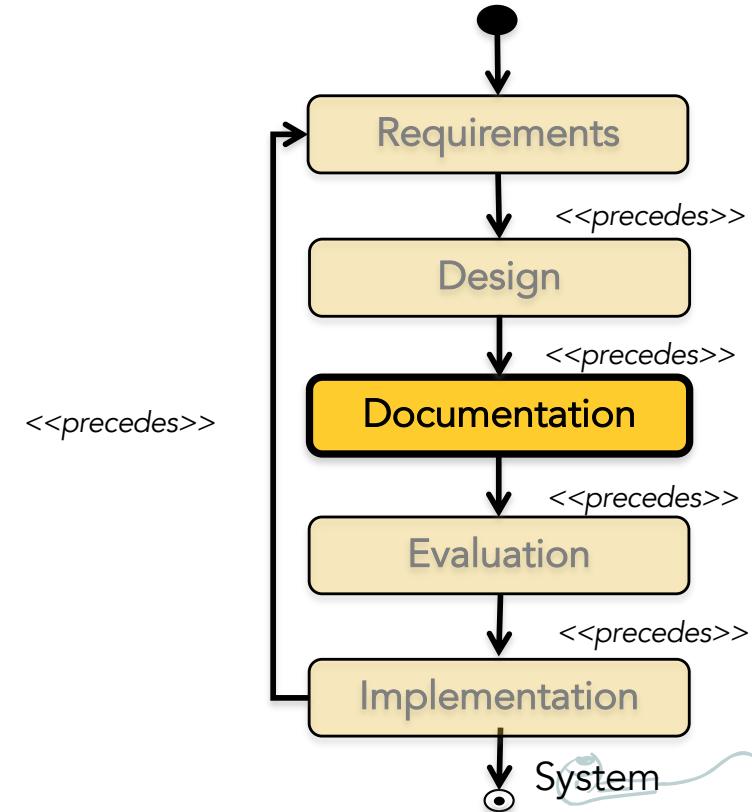
A **translation**, from the world
of needs (requirements) **to the**
world of solutions, in terms
of architectural structures.



Software Architecture Development Cycle

Communicating the architecture

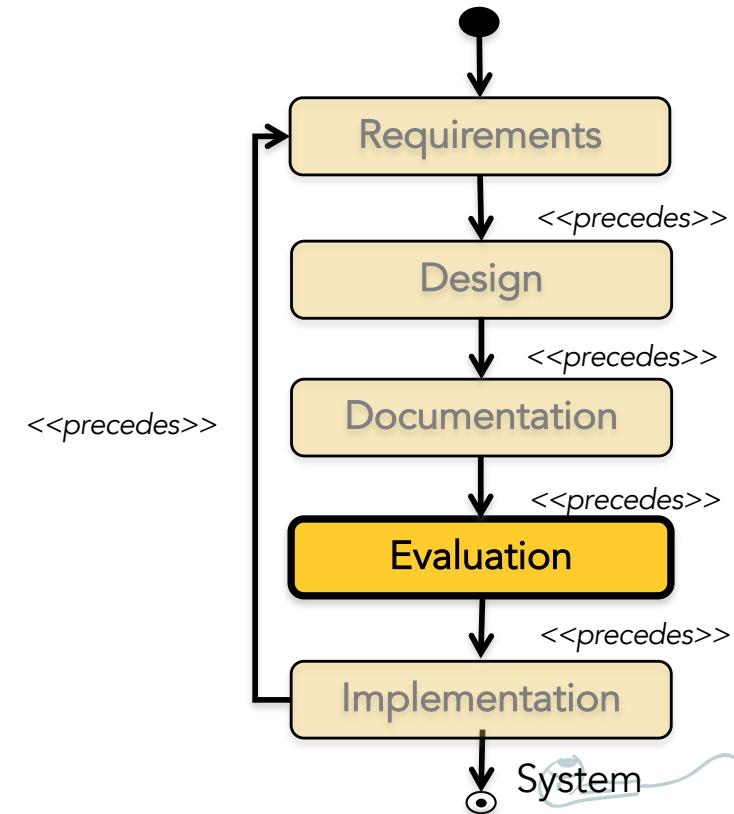
If a system is **big enough** and it is is **nontrivial**, it should be documented. In other engineering disciplines "**a blueprint**"--some sort of documented design, is an absolutely essential step in moving towards implementation and the commitment of resources.



Software Architecture Development Cycle

Verifying design decisions

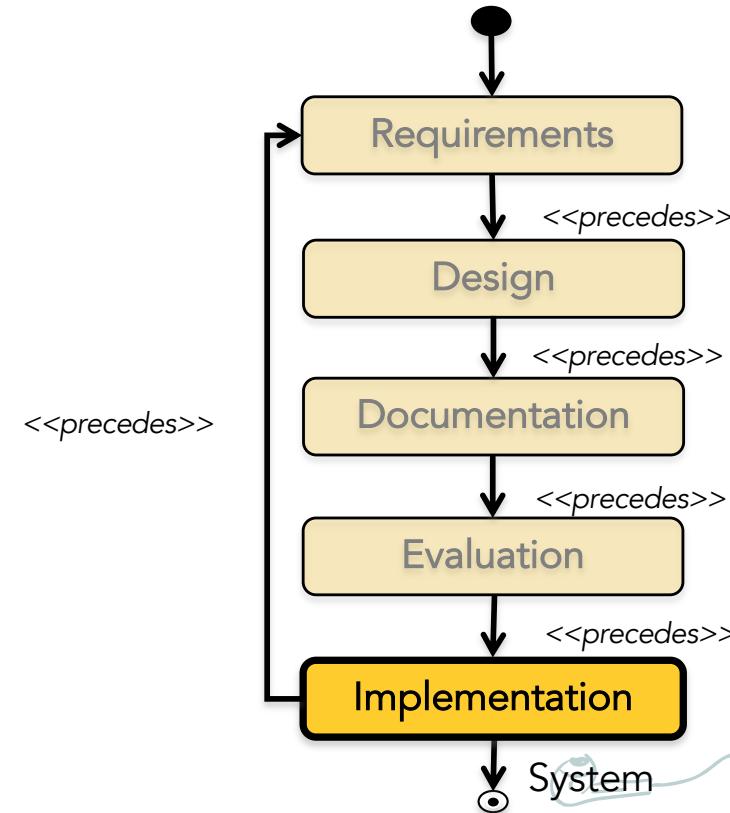
If your system is nontrivial, then you owe it to yourself and you owe it to your stakeholders to evaluate it, that is to ensure that the decisions made are **appropriate** to address the critical requirements.



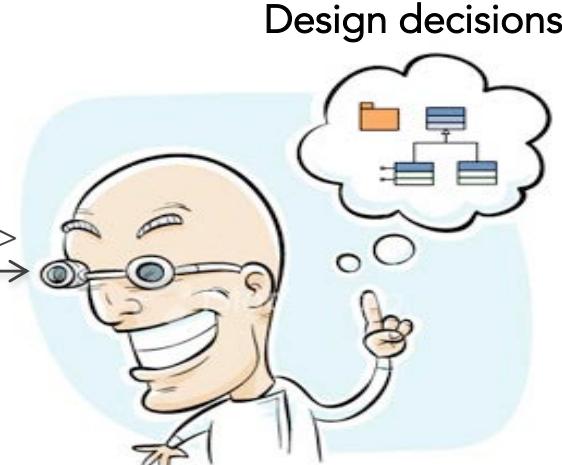
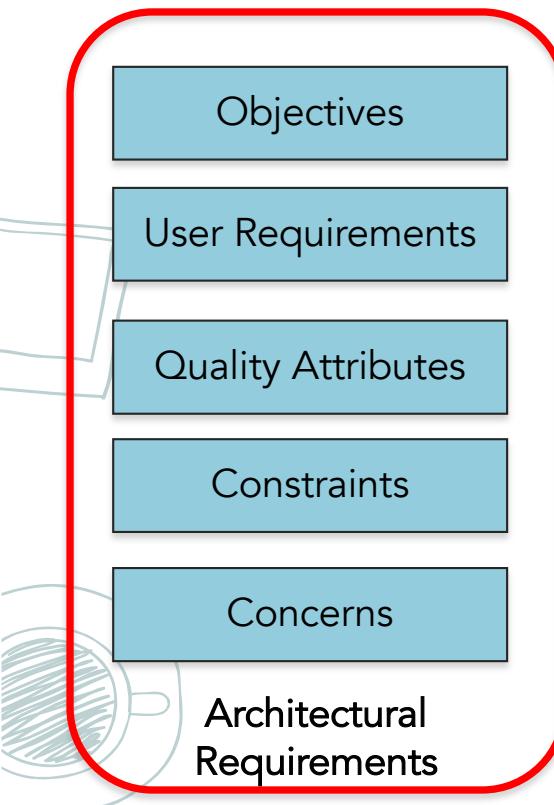
Software Architecture Development Cycle

Making it real

Finally, you need to **implement** the architecture that you have created (and evaluated).

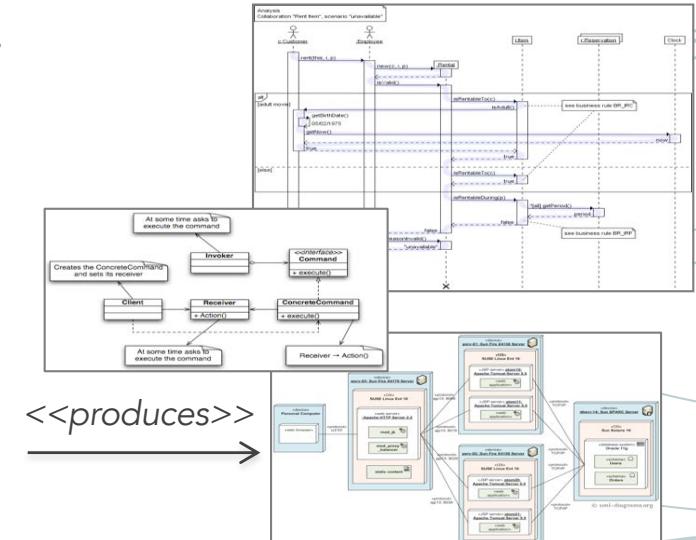


Software Architecture Development Cycle



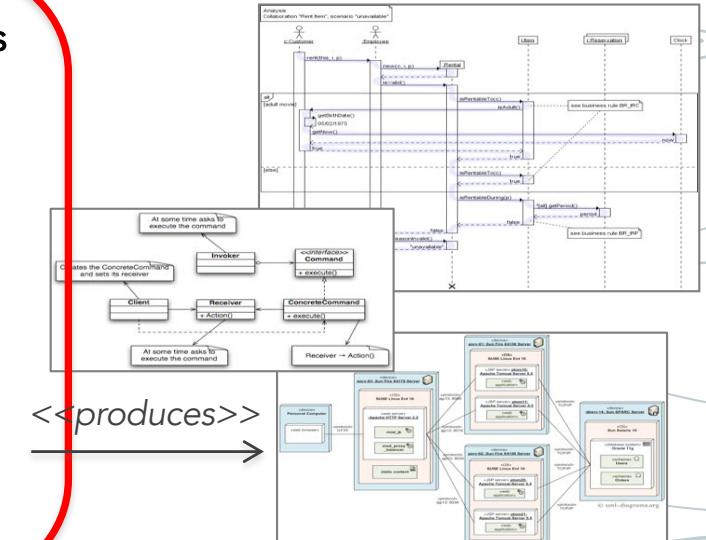
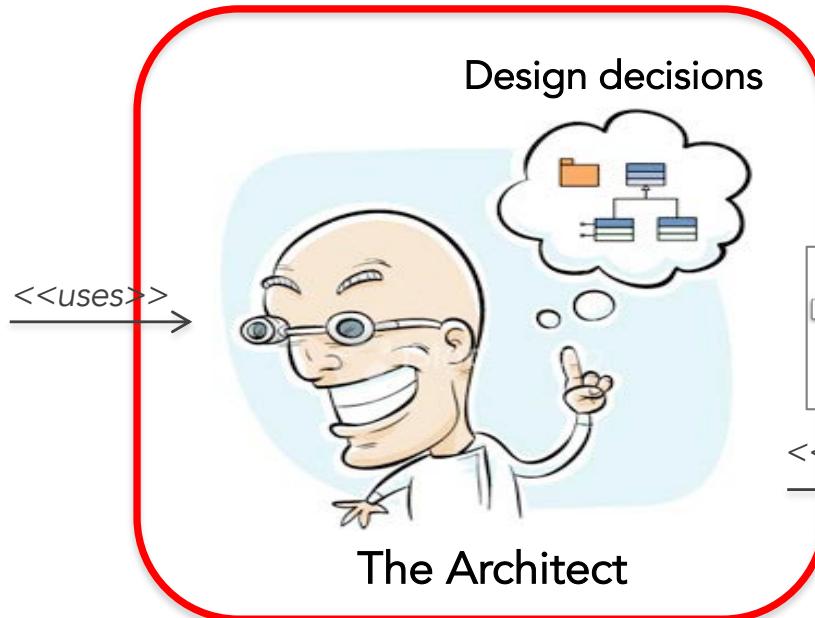
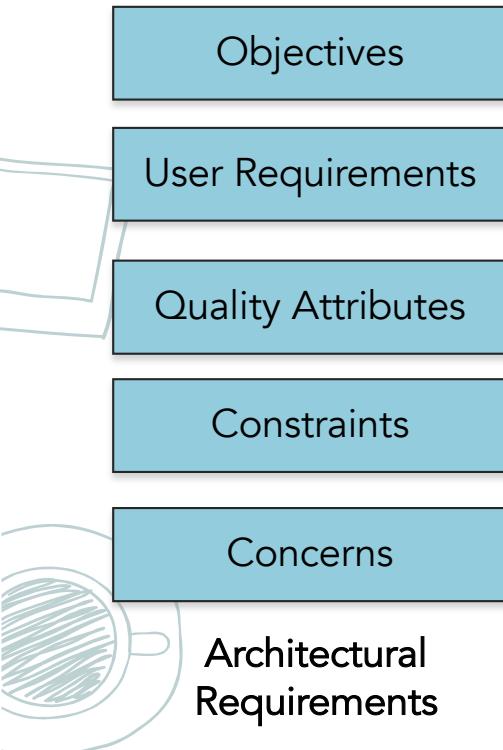
Design decisions

The Architect



Architectural
Structures

Software Architecture Development Cycle



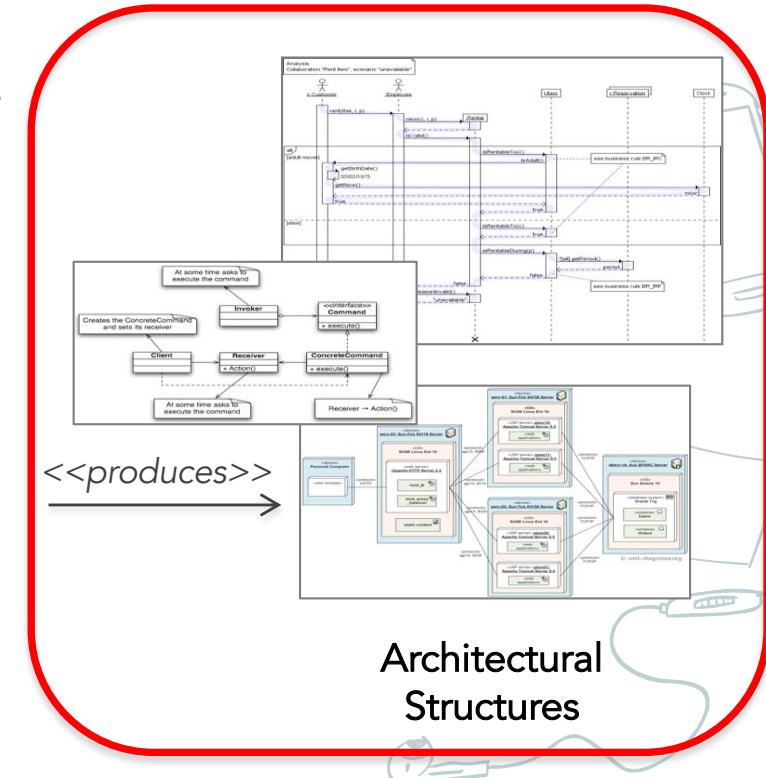
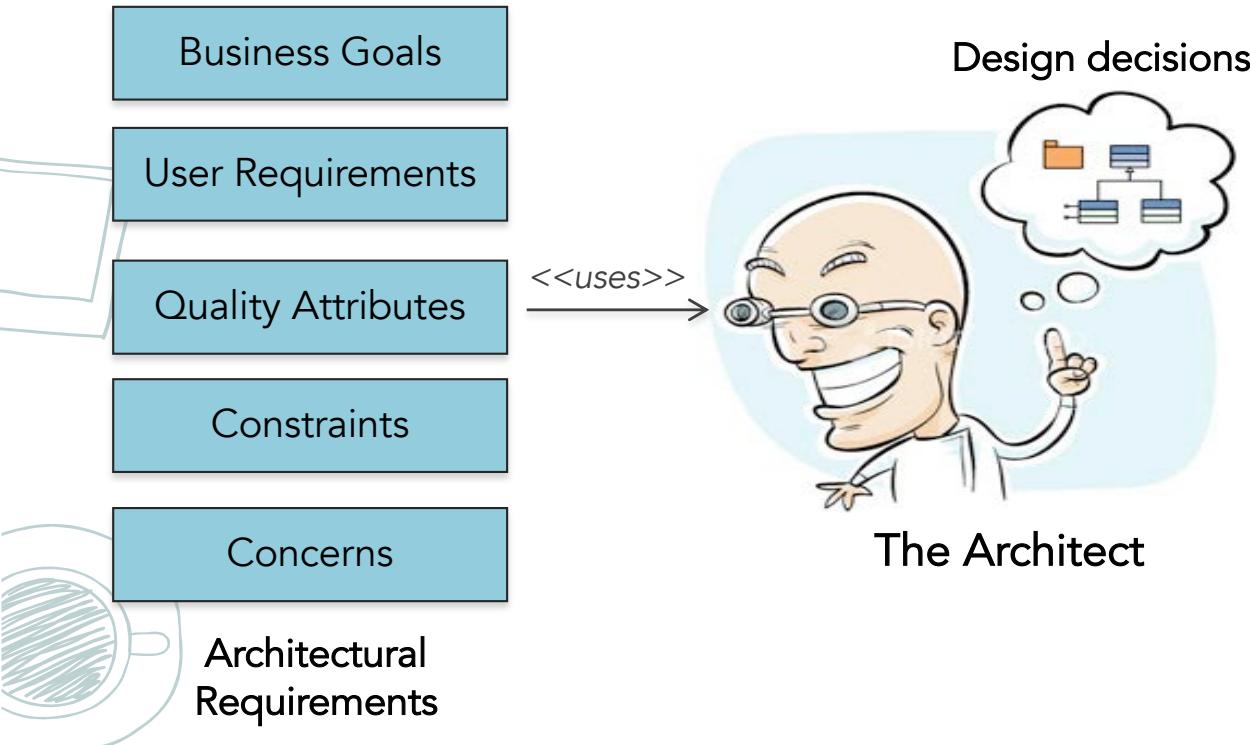
Architectural
Structures

Design Concepts

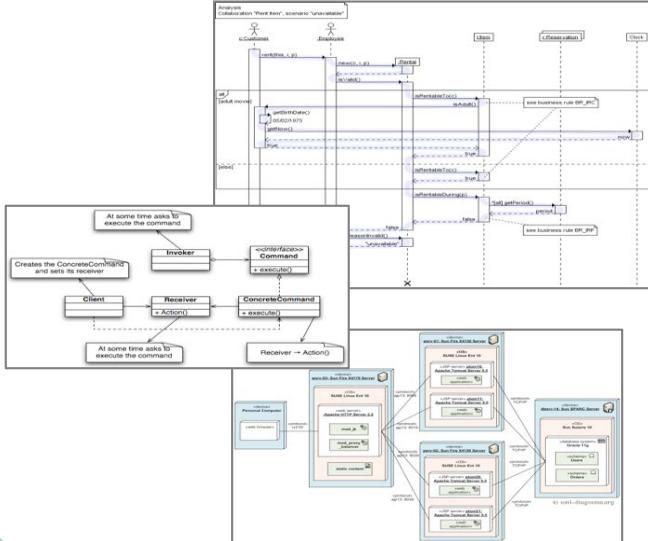
- Reference Architectures
- Patterns
- Tactics
- Technologies



Software Architecture Development Cycle

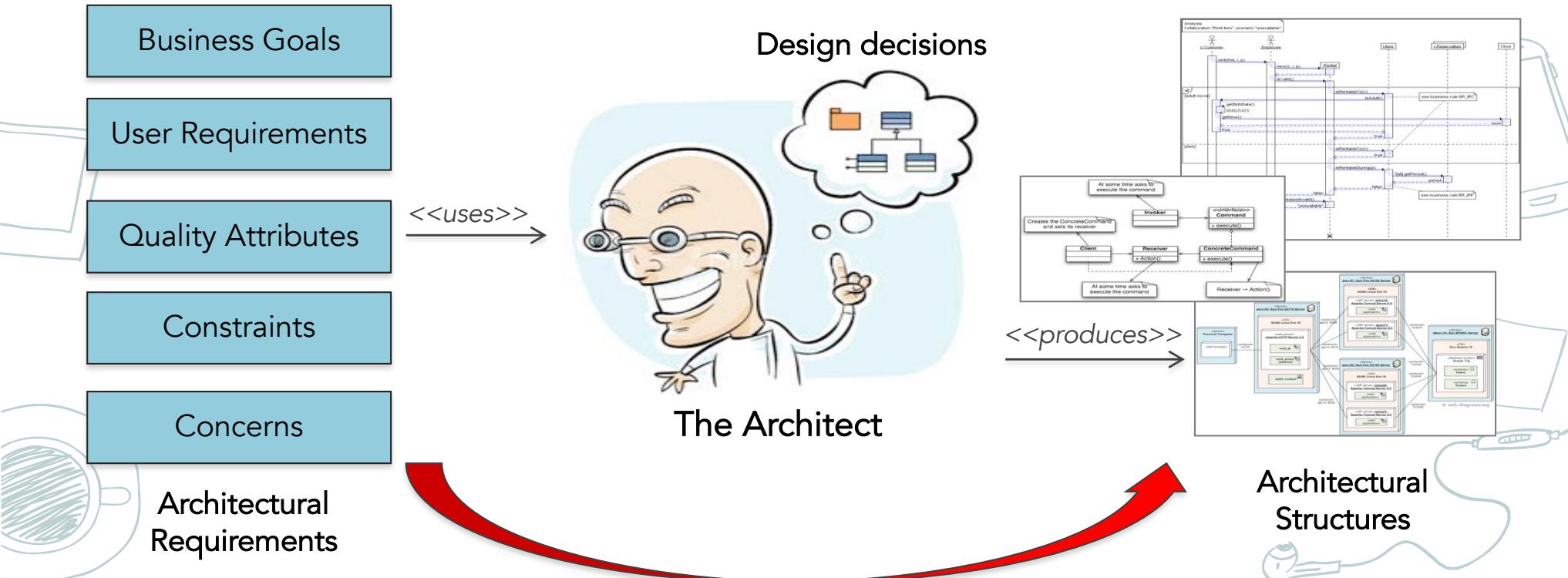


Notations



- Boxes and lines
- UML
- C4
- AADL
- ...

Software Architecture Development Cycle



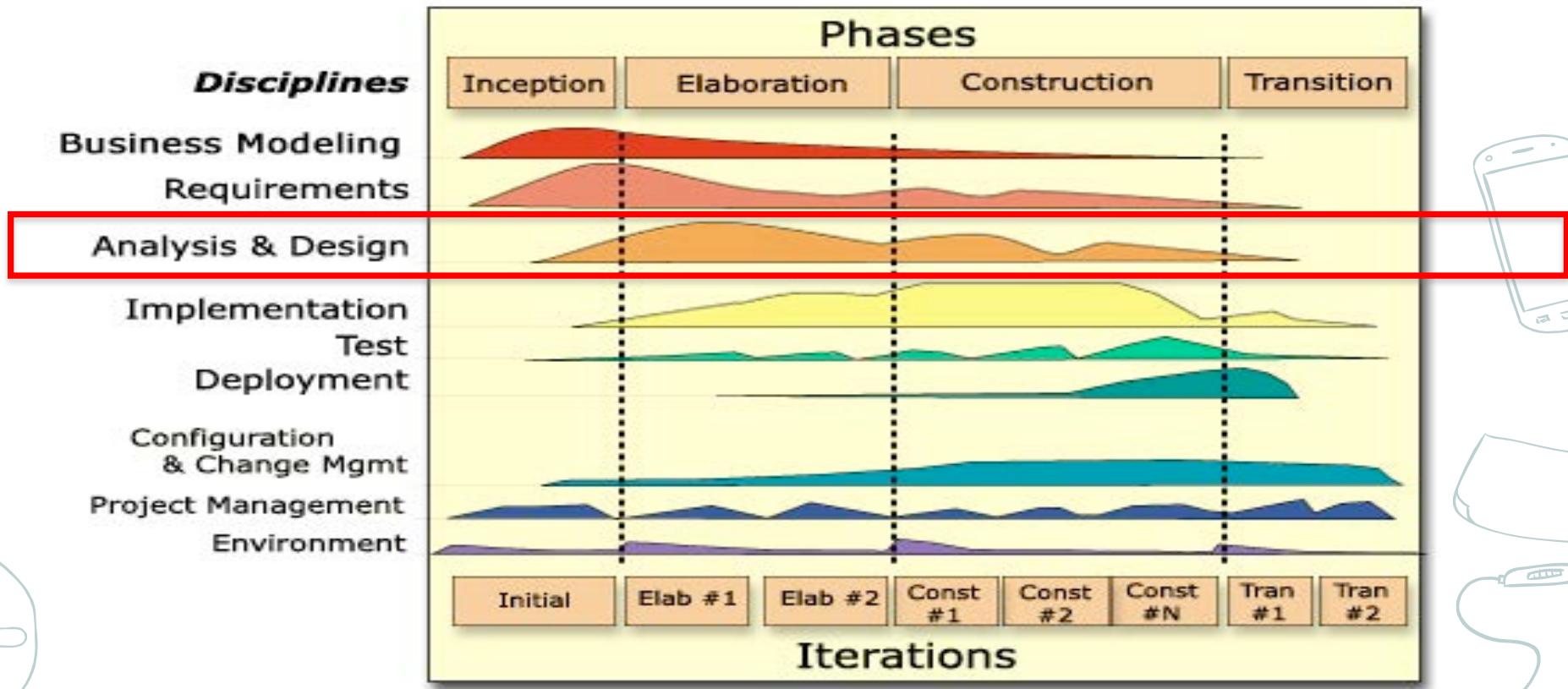
Outline

1. Software Life Cycle
2. Software Architecture Development Cycle
3. Architecture Driven Development
4. Summary

What is a Architecture Driven Development?

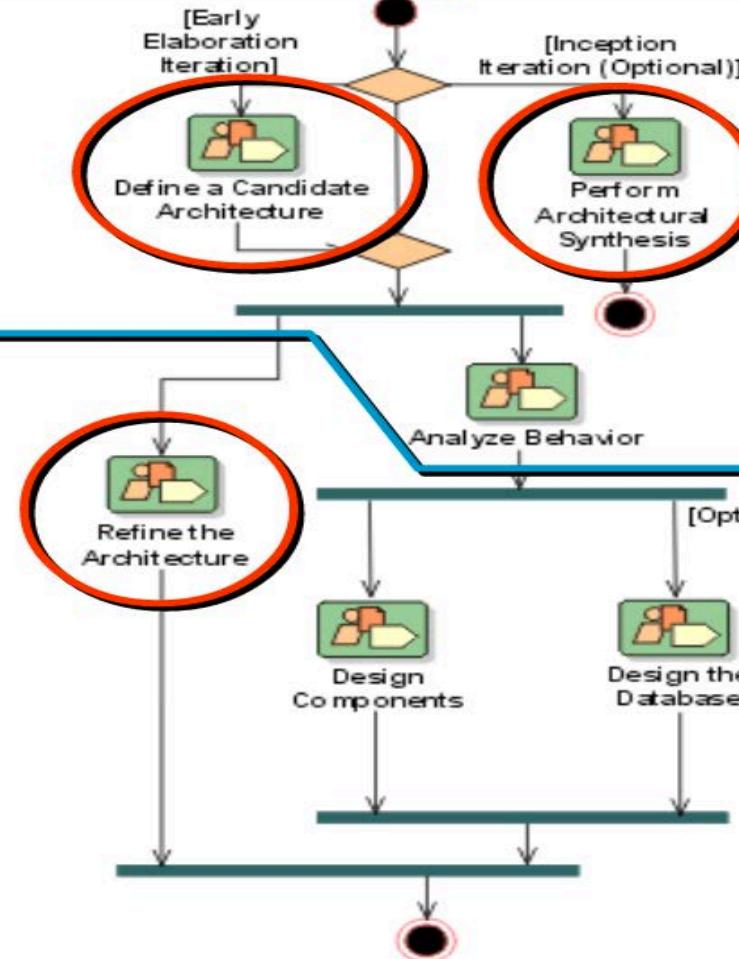


RUP



Analysis

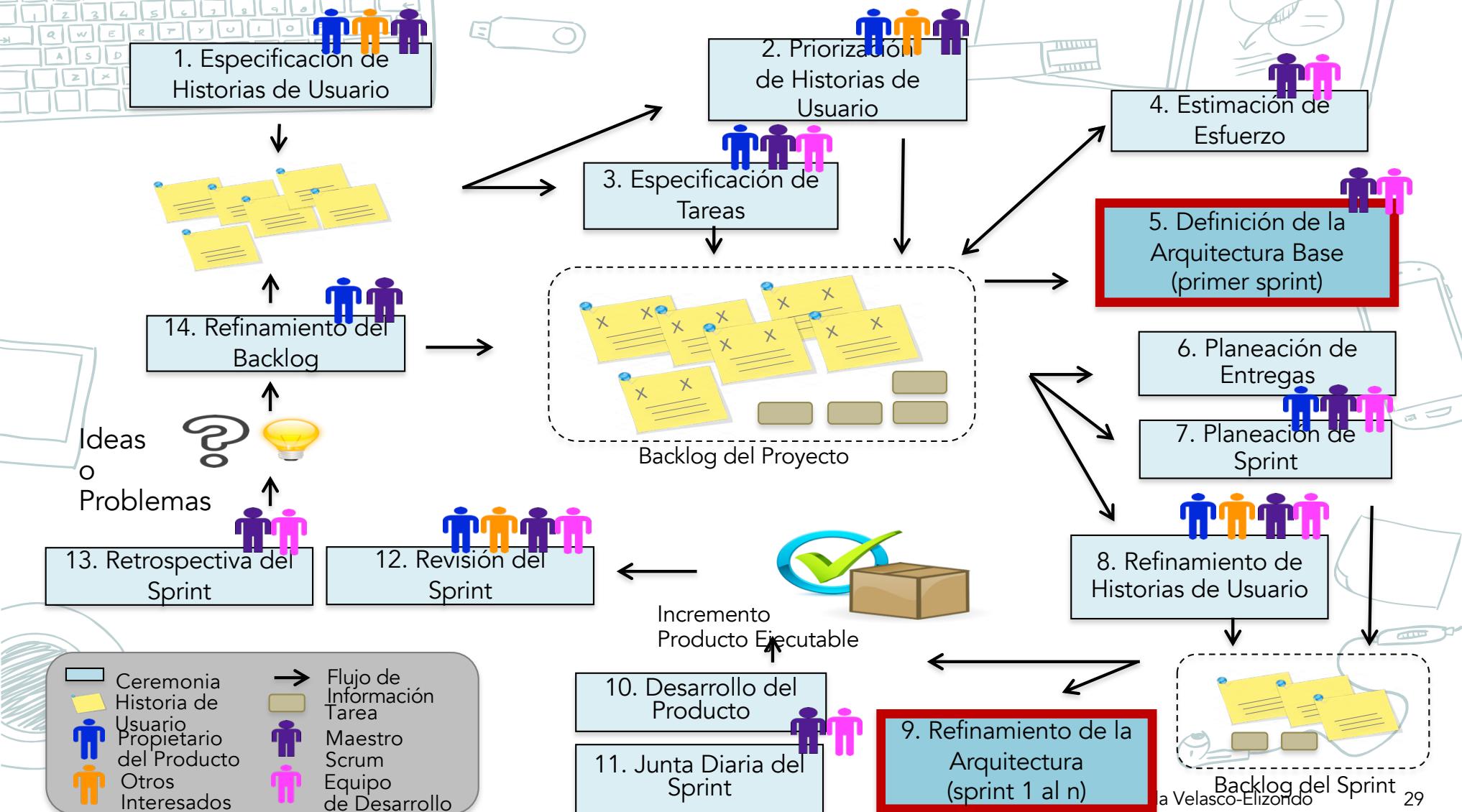
Design

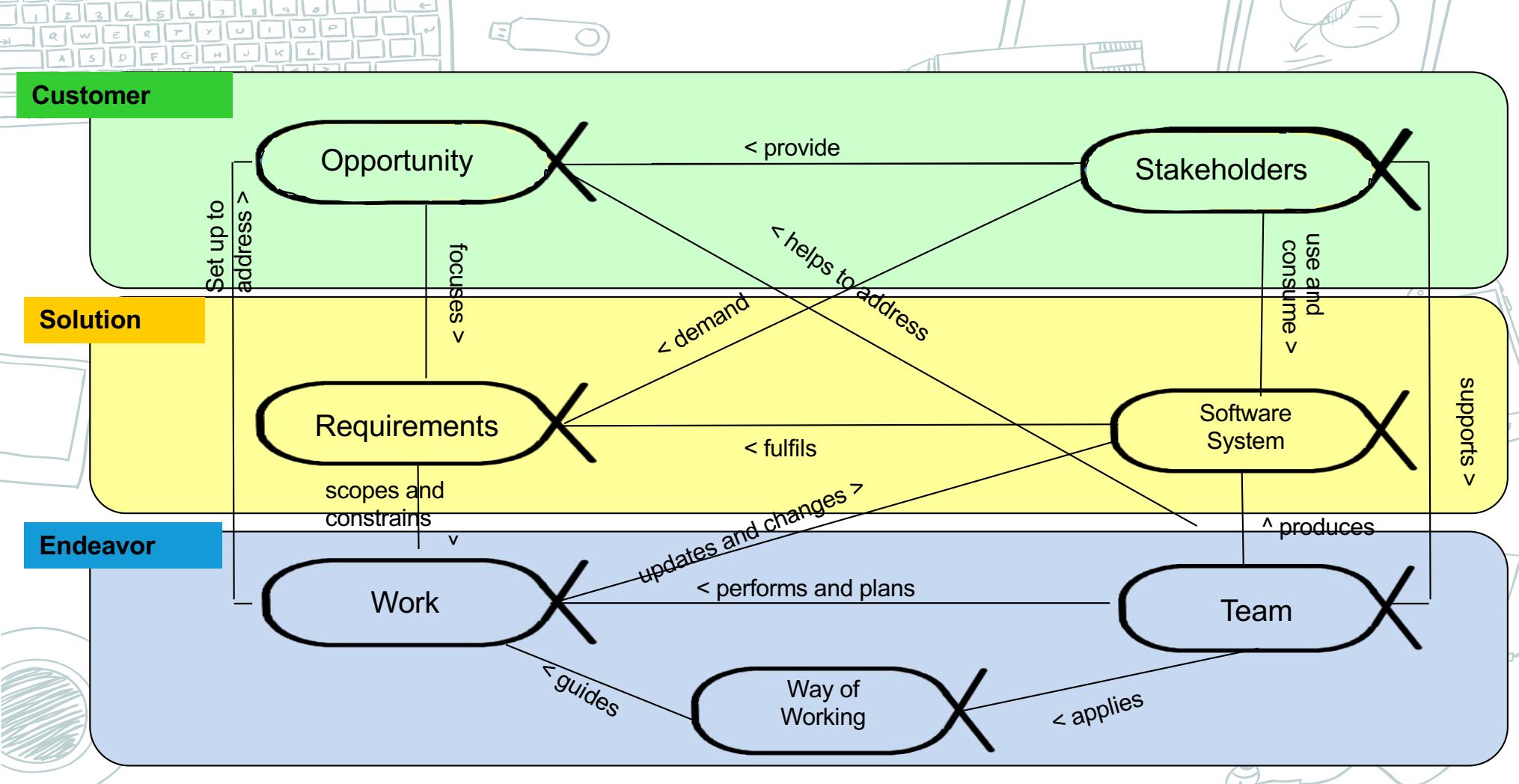


Scrum

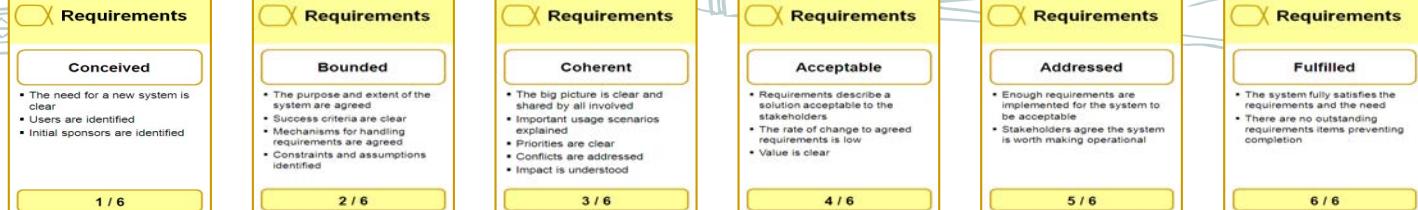


COPYRIGHT © 2005, MOUNTAIN GOAT SOFTWARE





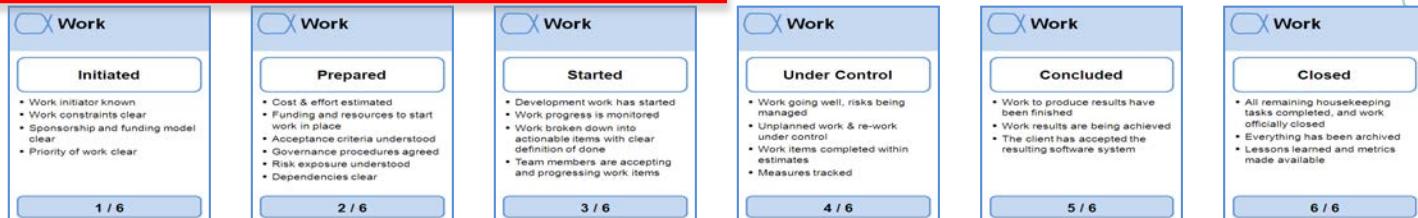
Requirements



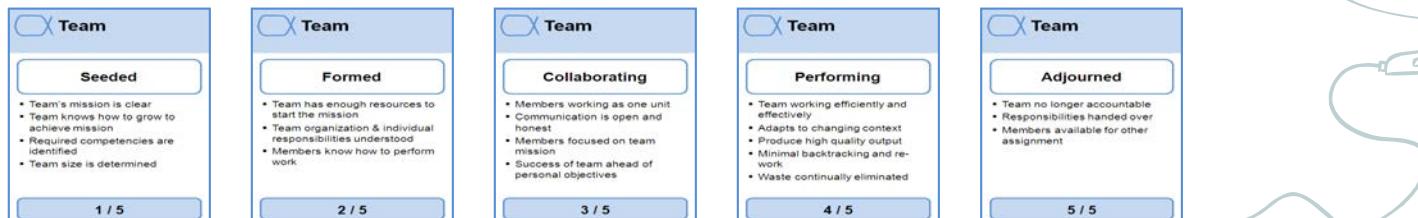
Software System



Work



Team



Outline

1. Software Life Cycle
2. Software Architecture Development Cycle
3. Architecture Centred Development
4. Summary

Summary

- Software Life Cycle
- Software Architecture Development Cycle
- Architecture Centred Development
- Summary

Questions?

Comments?

