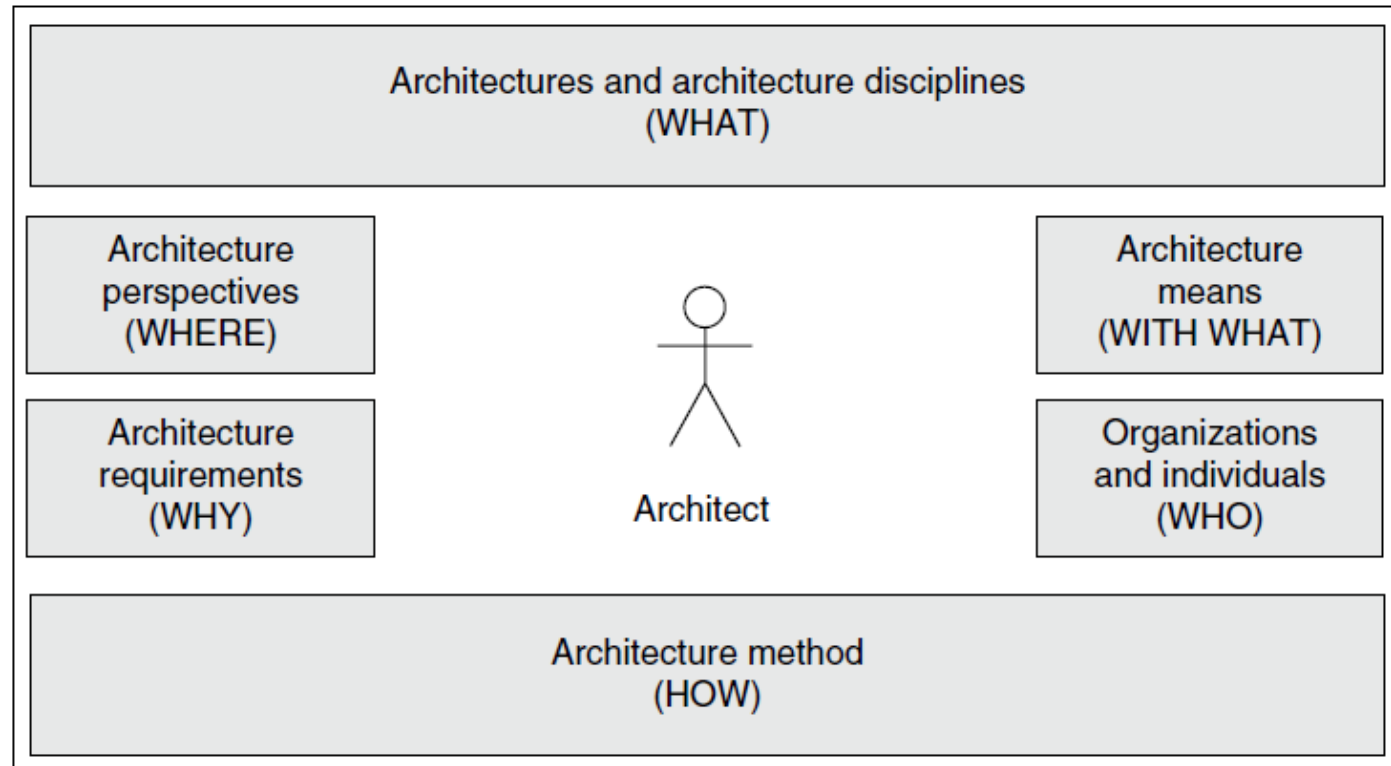


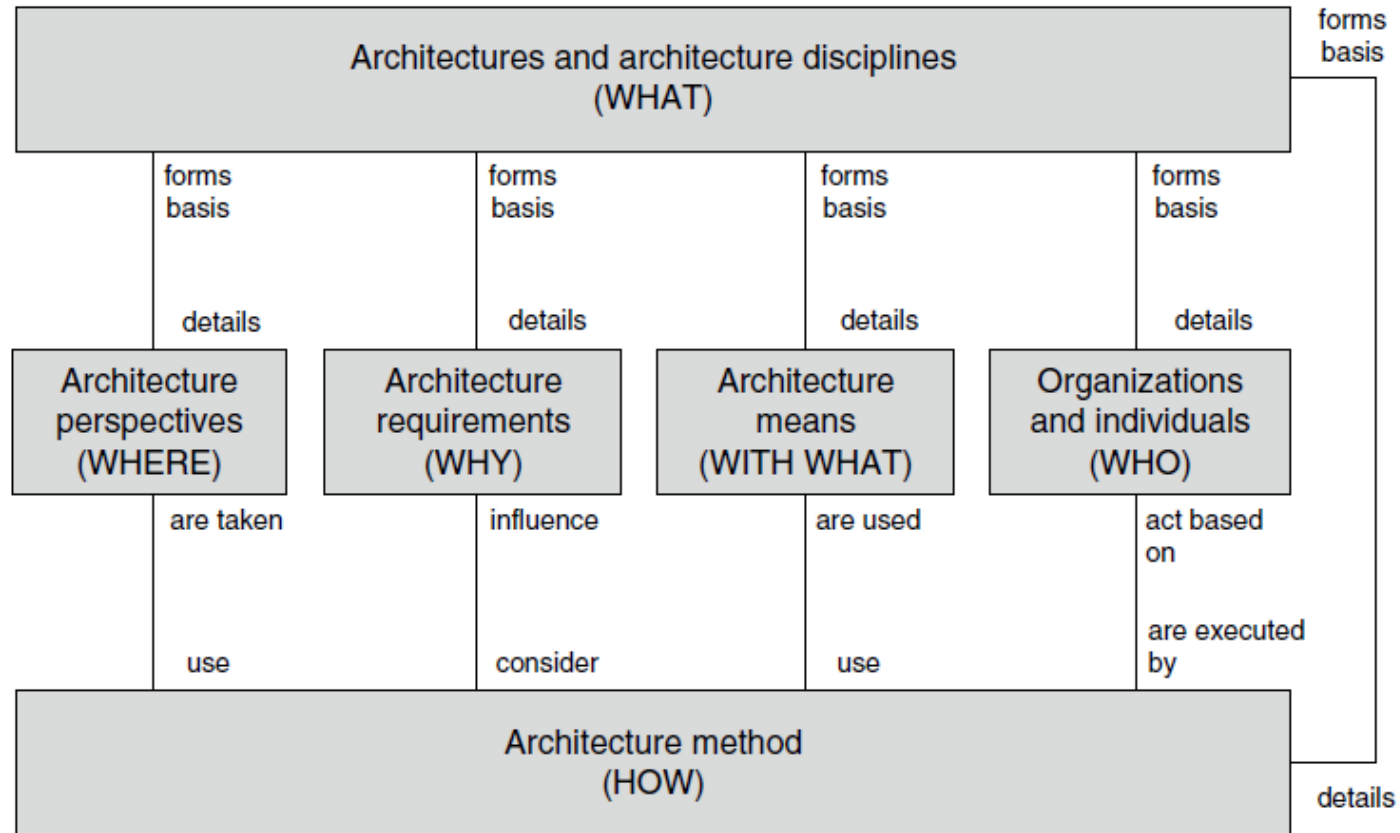



OOAD: Architectural Design
Presenter: Dr. Ha Viet Uyen Synh.

Overview of the architecture orientation framework



Relationships between the dimensions





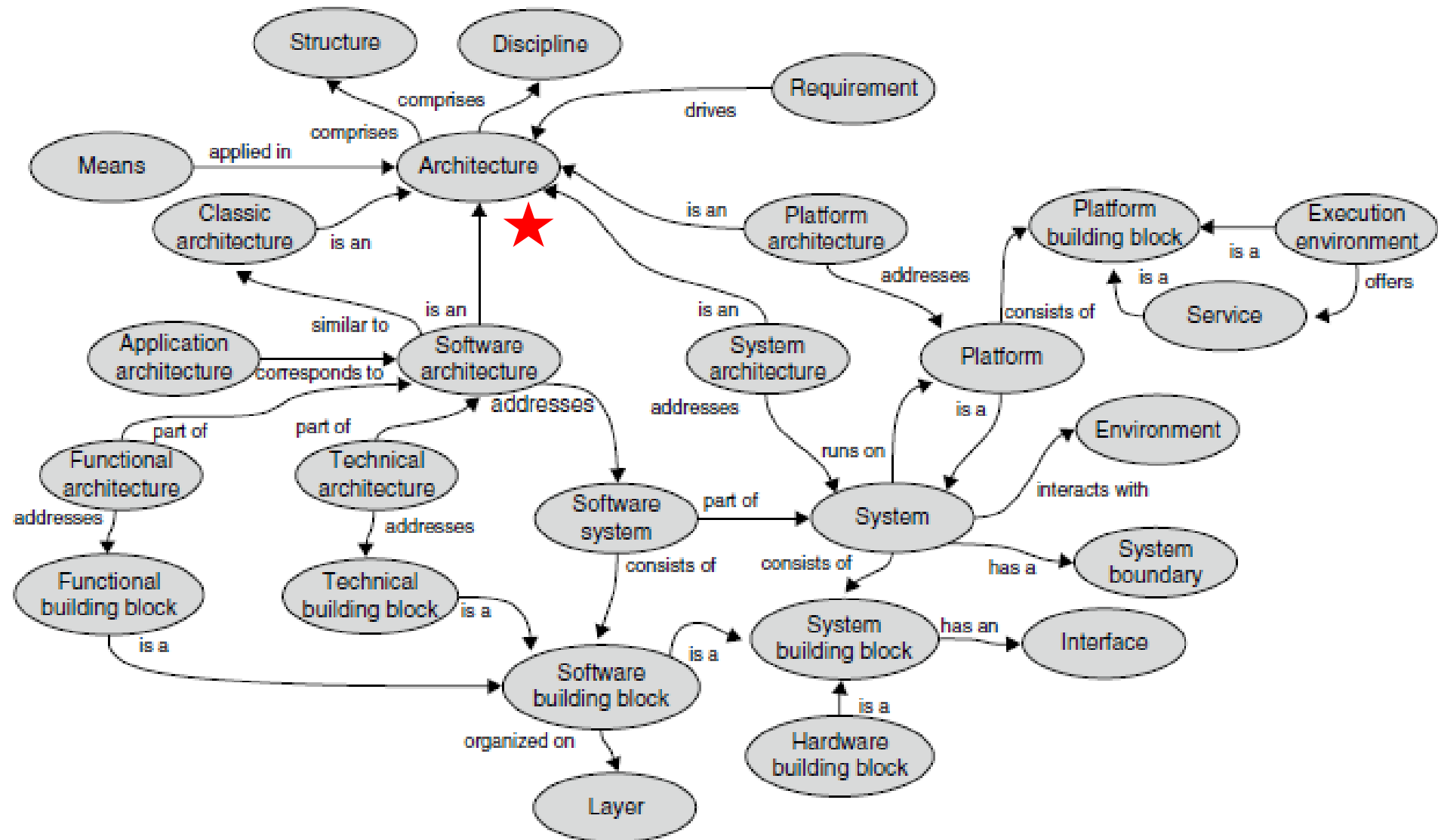
I. Architectures and Architecture Disciplines (WHAT)

The WHAT dimension (architectures and architecture disciplines) contains **basic principles and definitions** of architecture. It therefore lays the basis for working as an architect.

Architecture disciplines:

- Software architecture
- Data architecture
- Integration architecture
- Network architecture
- Security architecture
- System management architecture
- Enterprise architecture

Basic concepts of the WHAT dimension

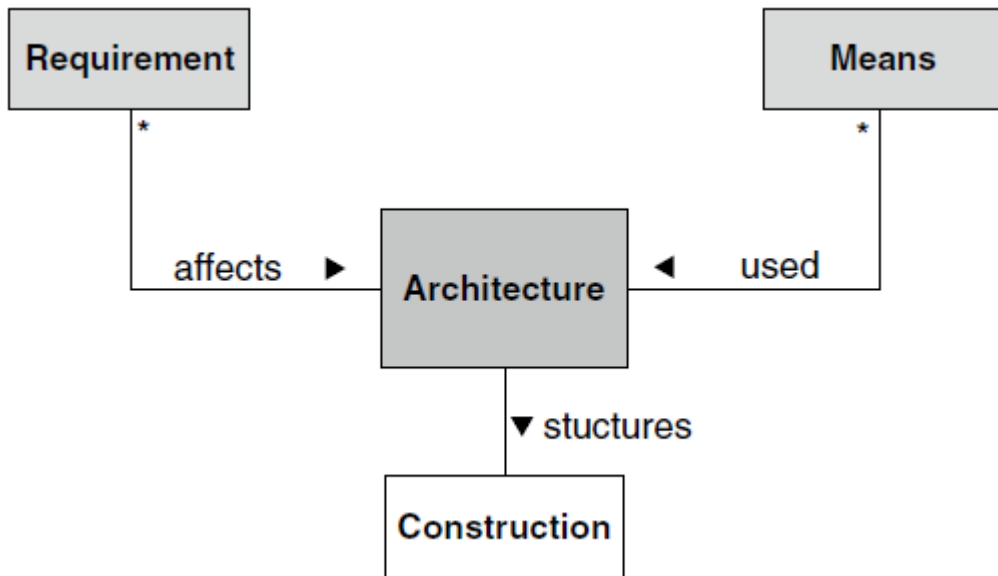


Classic Architecture as Starting Point

A classic architecture **comprises** structures and activities.

The architect communicates with **stakeholders**.

The architect uses different **perspectives**.

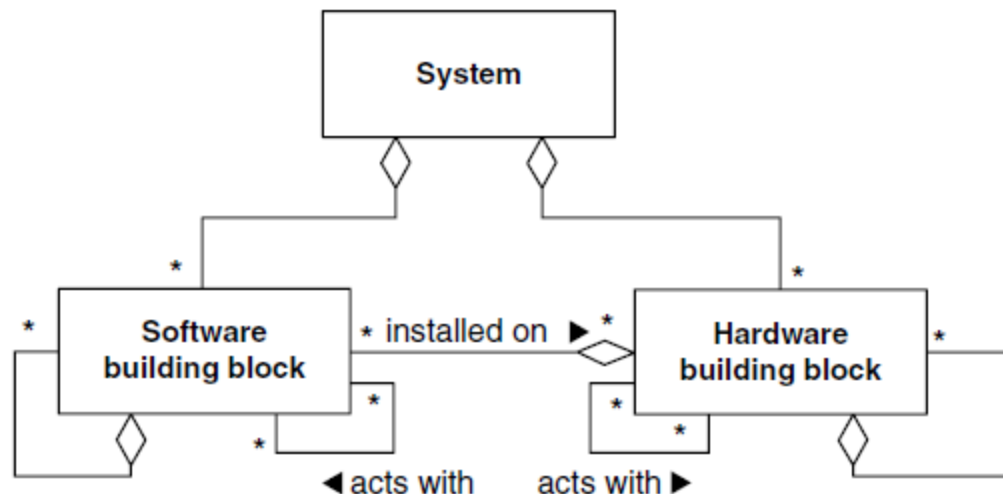


The architecture therefore defines the **fundamental parts**, but not the details of the system to be developed

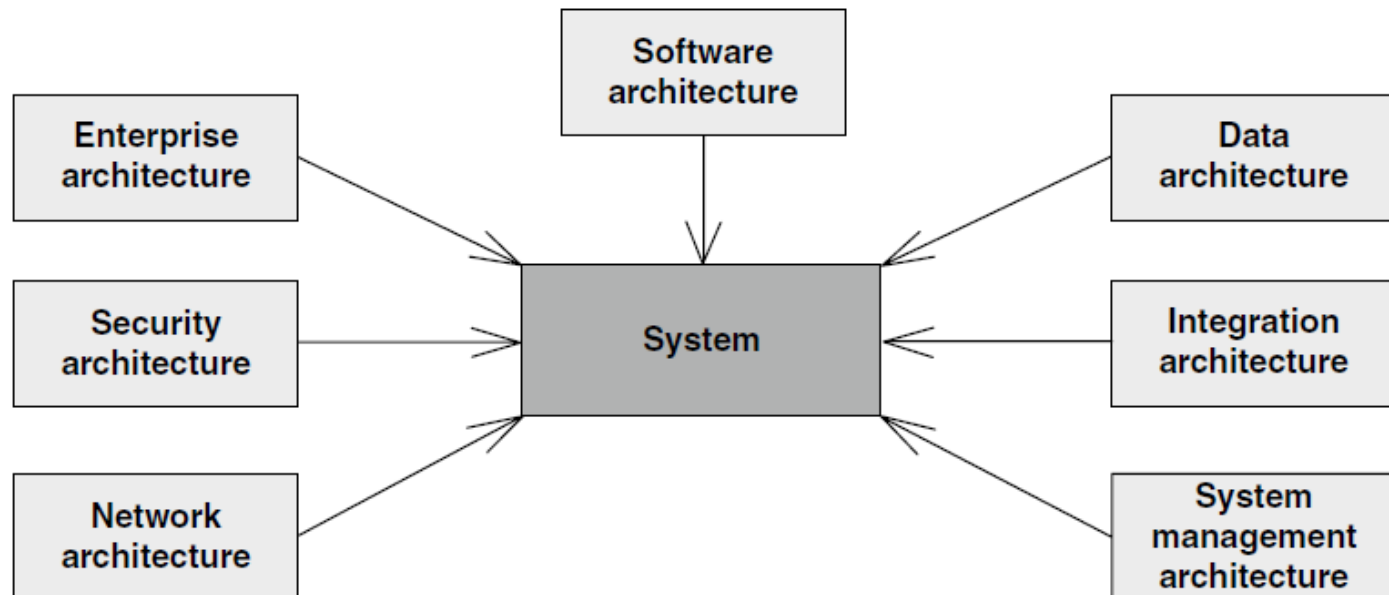
```
classDiagram
    class System
    class SoftwareBuildingBlock["Software building block"]
    class HardwareBuildingBlock["Hardware building block"]
    System "1" *-- "*" SoftwareBuildingBlock
    System "1" *-- "*" HardwareBuildingBlock
    SoftwareBuildingBlock "*" *-- "*" HardwareBuildingBlock : installed on
    SoftwareBuildingBlock "*" *-- "*" SoftwareBuildingBlock : acts with
    HardwareBuildingBlock "*" *-- "*" HardwareBuildingBlock : acts with
```

The diagram illustrates the relationships between three classes: **System**, **Software building block**, and **Hardware building block**.

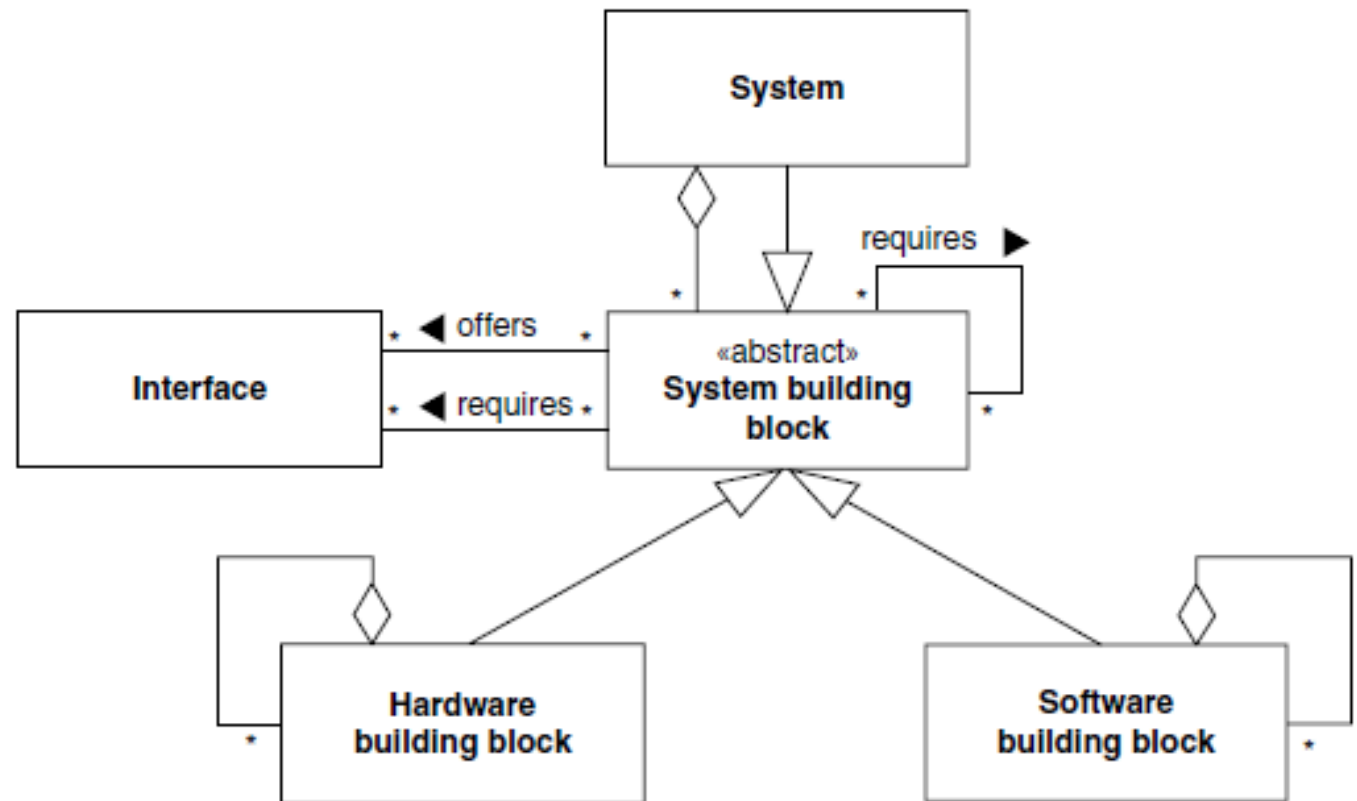
- System** is associated with **Software building block** (multiplicity 1 to *) and **Hardware building block** (multiplicity 1 to *) via composition (diamonds).
- Software building block** is associated with **Hardware building block** (multiplicity * to *) via the relationship **installed on** (arrow with open diamond).
- Software building block** has a self-association (multiplicity * to *) labeled **acts with** (arrow with open diamond).
- Hardware building block** has a self-association (multiplicity * to *) labeled **acts with** (arrow with open diamond).



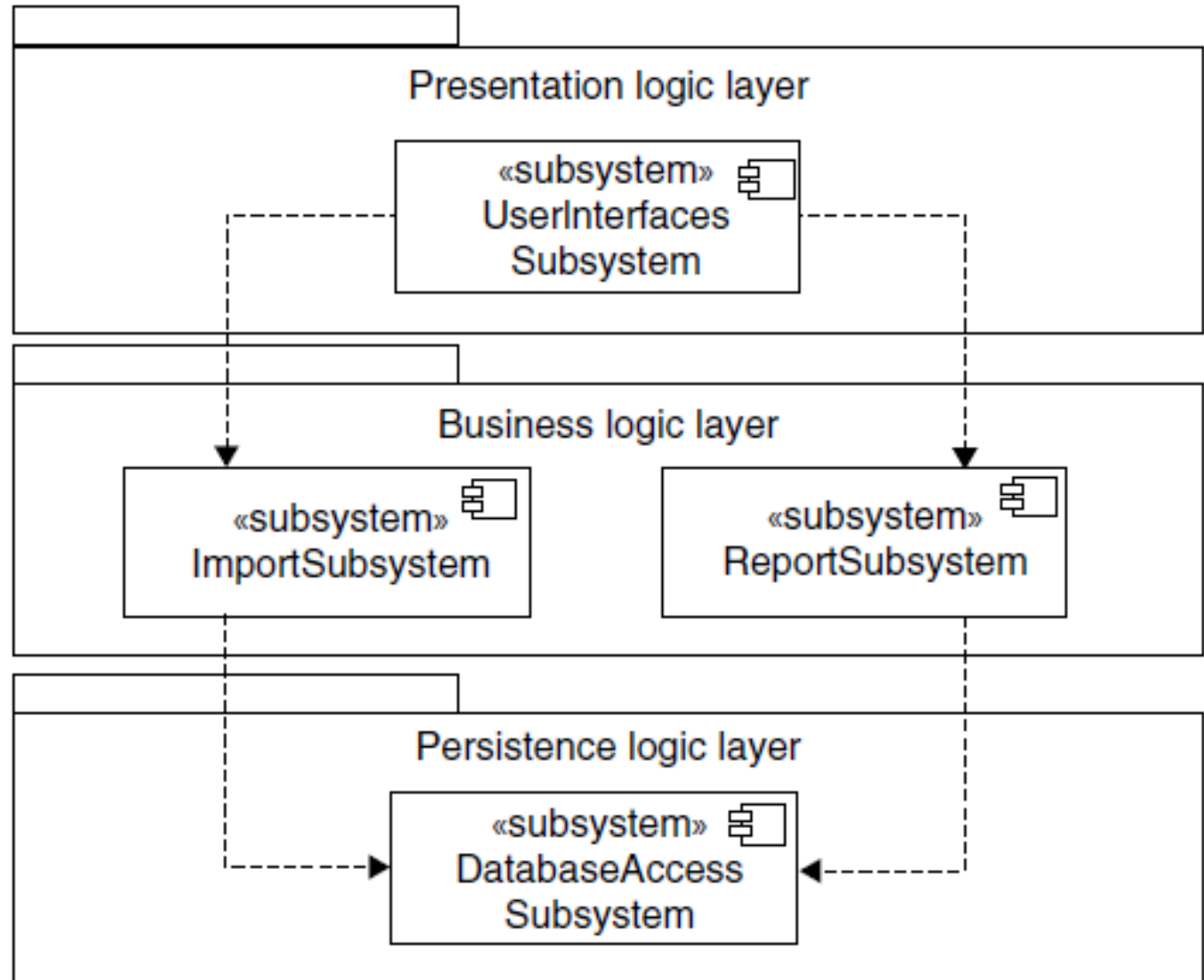
Interaction of architecture disciplines



System building blocks



Example





II. Architecture Perspectives (WHERE)

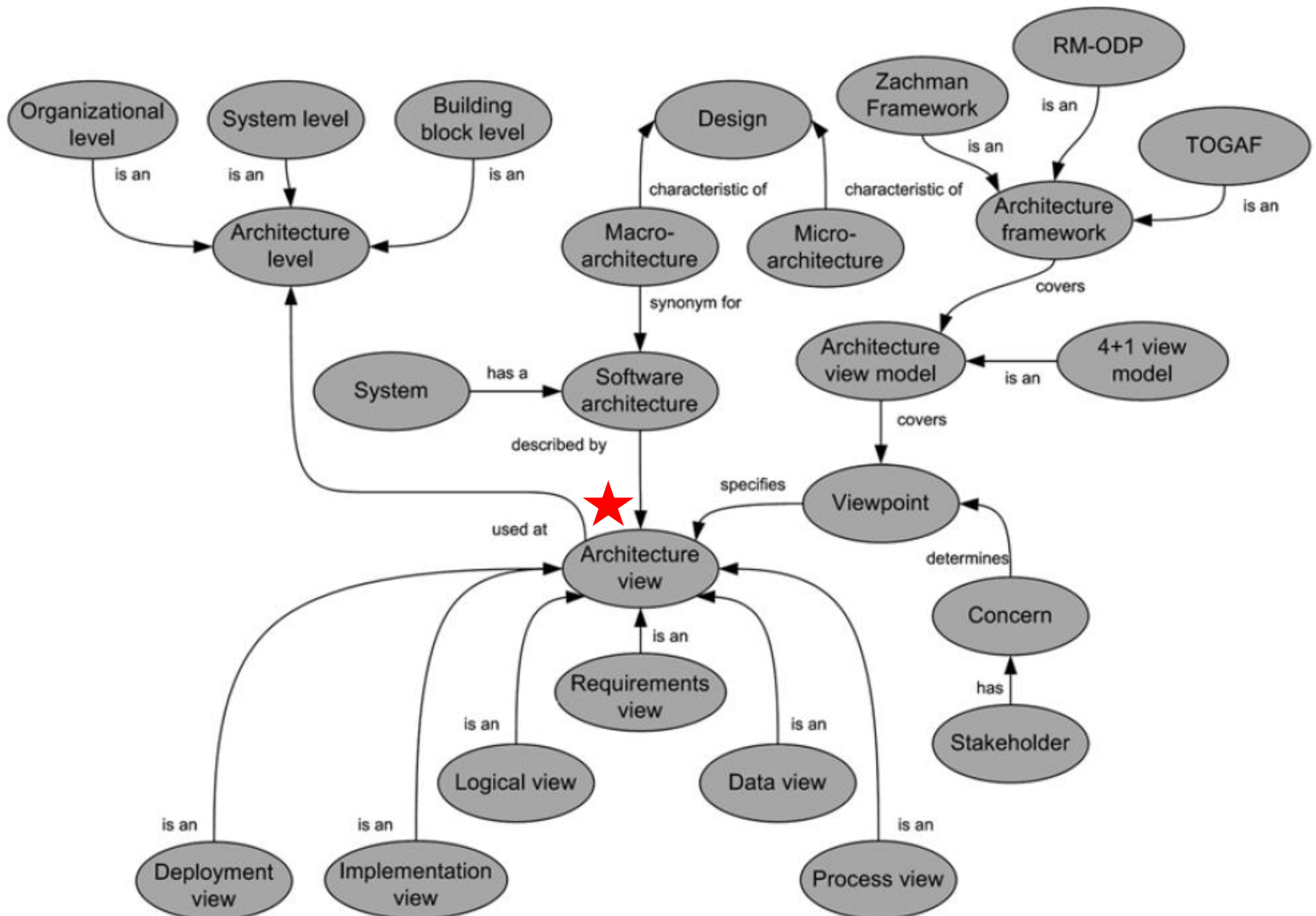
How the building blocks of a system are grouped?

How they interact?

How they are distributed and their behavior at runtime all at once?.

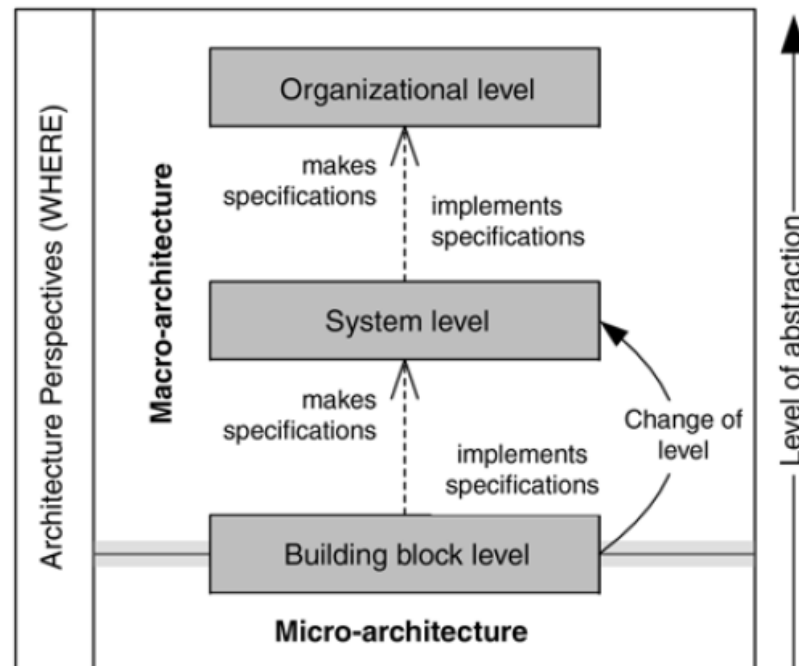
To be successful despite the restrictions of human understanding, you have **to reduce the complexity** by examining only one manageable part of an architecture at any one time.

Basic concepts of the WHERE dimension



Architecture Levels

- Organizational level: Here you can look at the organizations (e.g., departments and IT standards).
- System level: Here you can look at the systems of the organizations (e.g., IT systems and their requirements).
- Building block level: Here you can look at the building blocks of the systems (e.g., interfaces and data access objects).

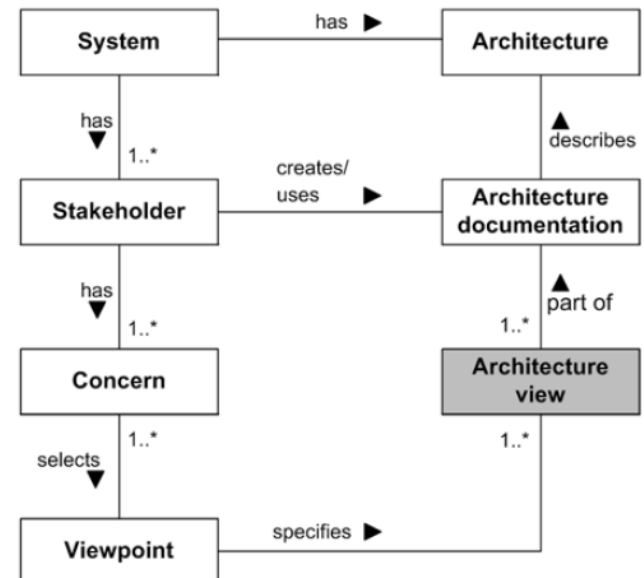


Architecture Views

Architecture views make **complex systems understandable**

Architecture views are motivated by stakeholders

- Stakeholders have different concerns with regard to a system, or rather, its architecture. Based on their specific concerns, different stakeholders each adopt a specific viewpoint.
- A specific architecture view (e.g., data view) is available to stakeholders from each respective viewpoint (different angles). It is defined by the viewpoint and refers to a set of specific concerns.
- Each architecture view has a reference to exactly one viewpoint.
- Architecture views essentially determine the content and structure of architecture documentation.

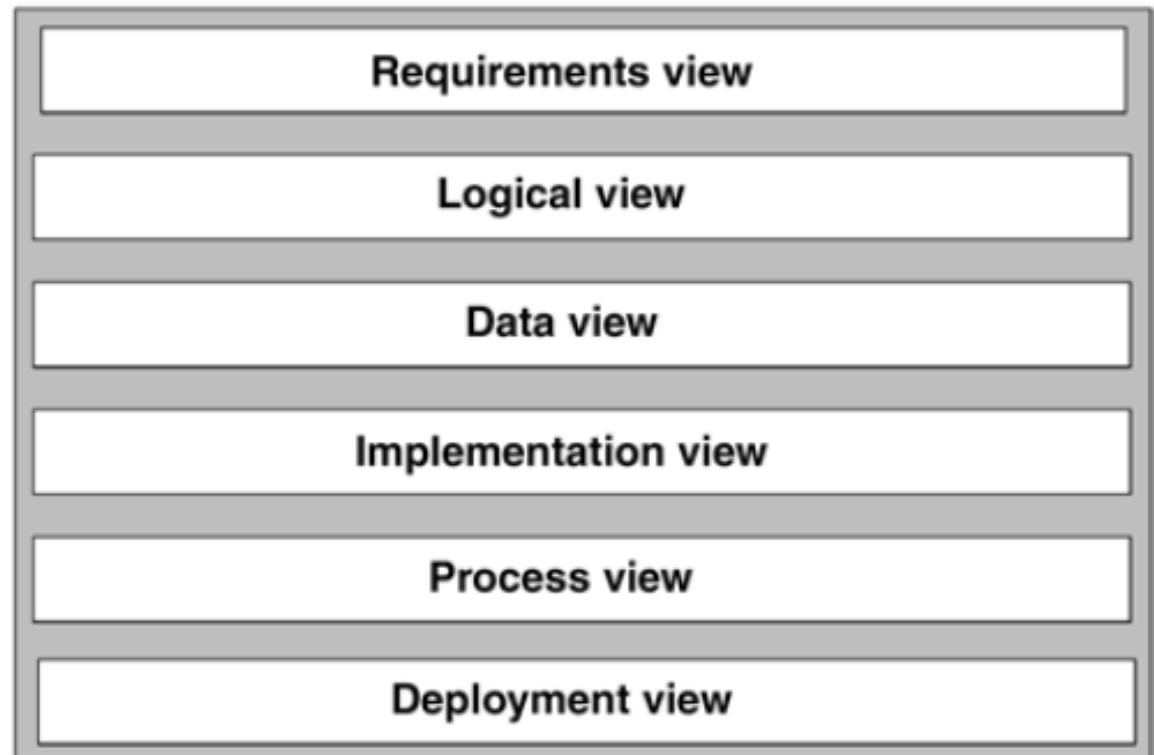




Basic architecture views

- **Conceptual view**: This architecture view describes the system building blocks and their relationships to one another *without going into detail* about, for example, interfaces. It is therefore suitable for conveying an architecture to non-technical stakeholders.
- **Logical view**: This architecture view describes the system building blocks and their *relationships to one another in detail*. The system building blocks and their relationships, or communication mechanisms, are specified precisely. This is necessary for the technical realization. This architecture view is therefore directed towards technical stakeholders. The conceptual view may be part of the logical view and therefore not represented explicitly. In this case the logical view partially addresses non-technical stakeholders.
- **Execution view**: This architecture view describes the *physical deployment* of the system building blocks at runtime in detail. It is also directed towards technical stakeholders.

Common architecture view model





III. Architecture Requirements (WHY)

Types of requirements

- Organizational requirements
- System requirements
- Building block requirements
- Development time requirements
- Runtime requirements
- Organizational constraints

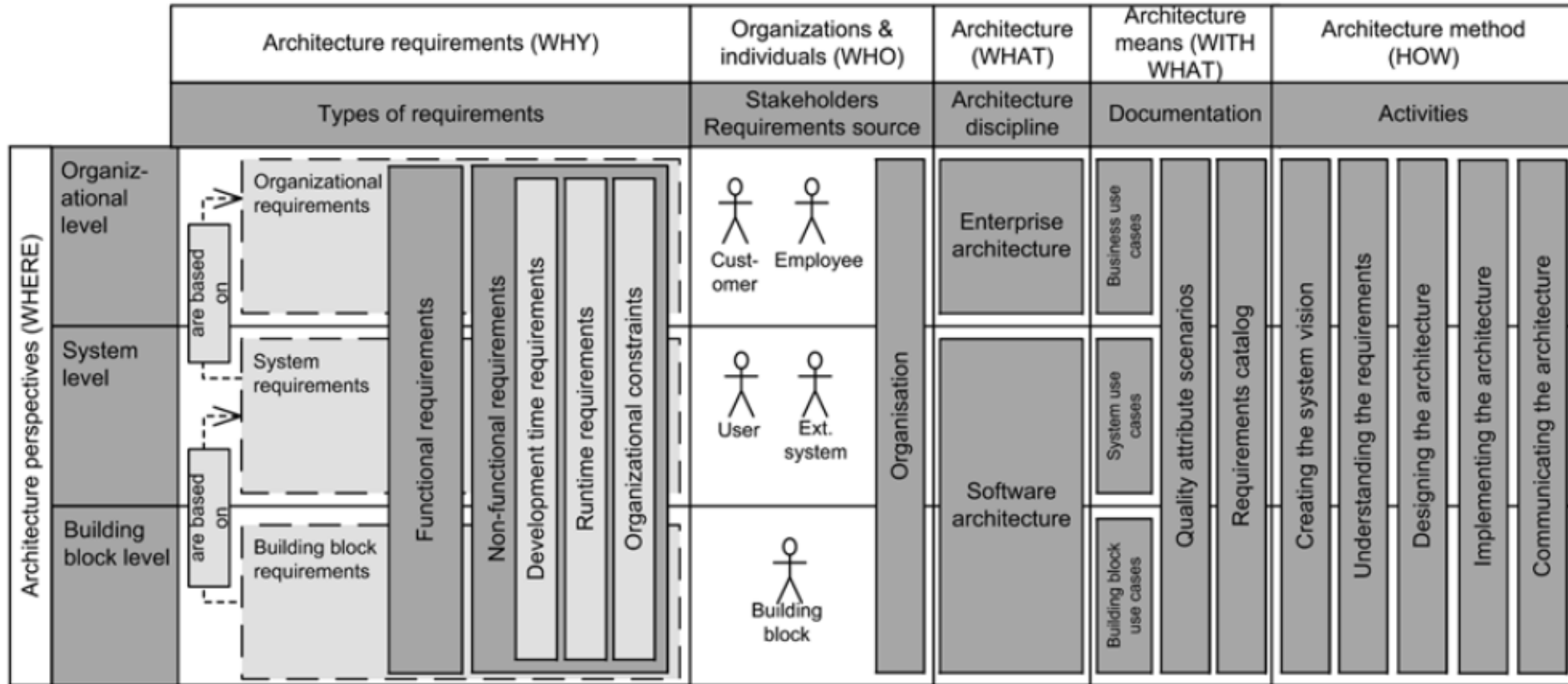
A requirement is:

- (1) A condition or capability needed by a user to solve a problem or achieve an objective.
- (2) A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed document.
- (3) A documented representation of a condition or capability as in (1) or (2).

Basic concepts of the WHY dimension



Requirements types





IV. Architecture Means (WITH WHAT)

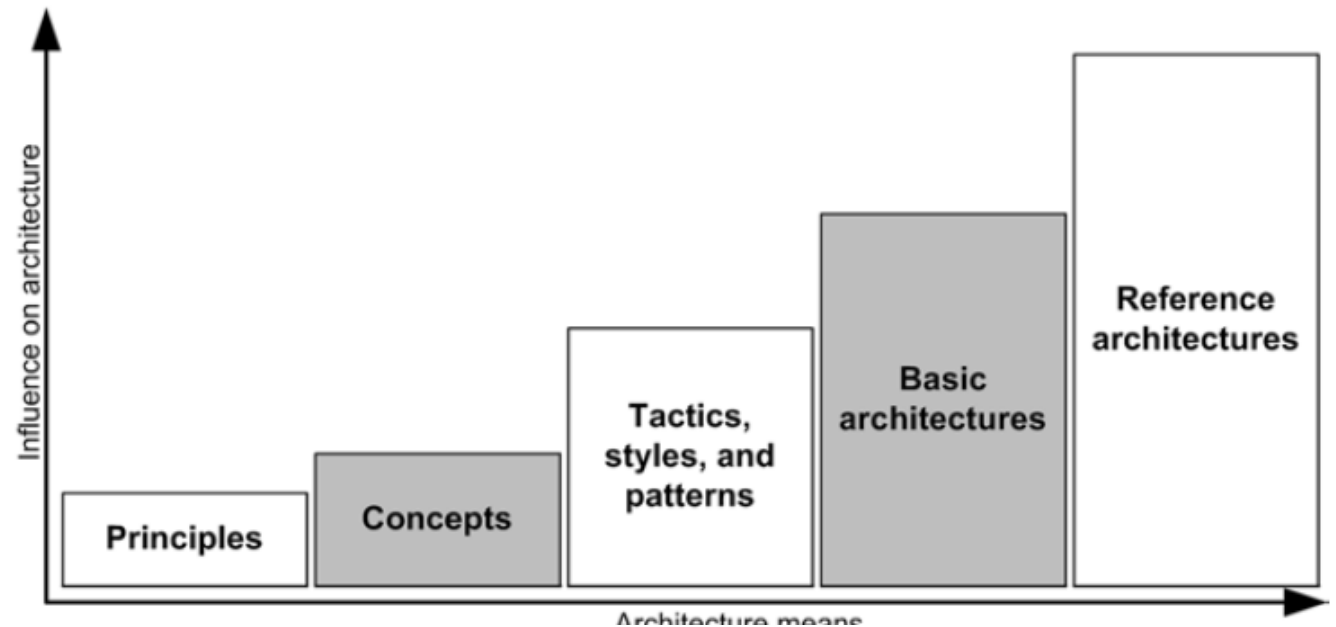
Basic architectures

- Cloud Computing Architectures
- Dataflow architecture
- Layered architecture
- Middleware architecture
- n-tier architecture
- Rich client architecture
- Service Oriented Architectures
- Thin client architecture

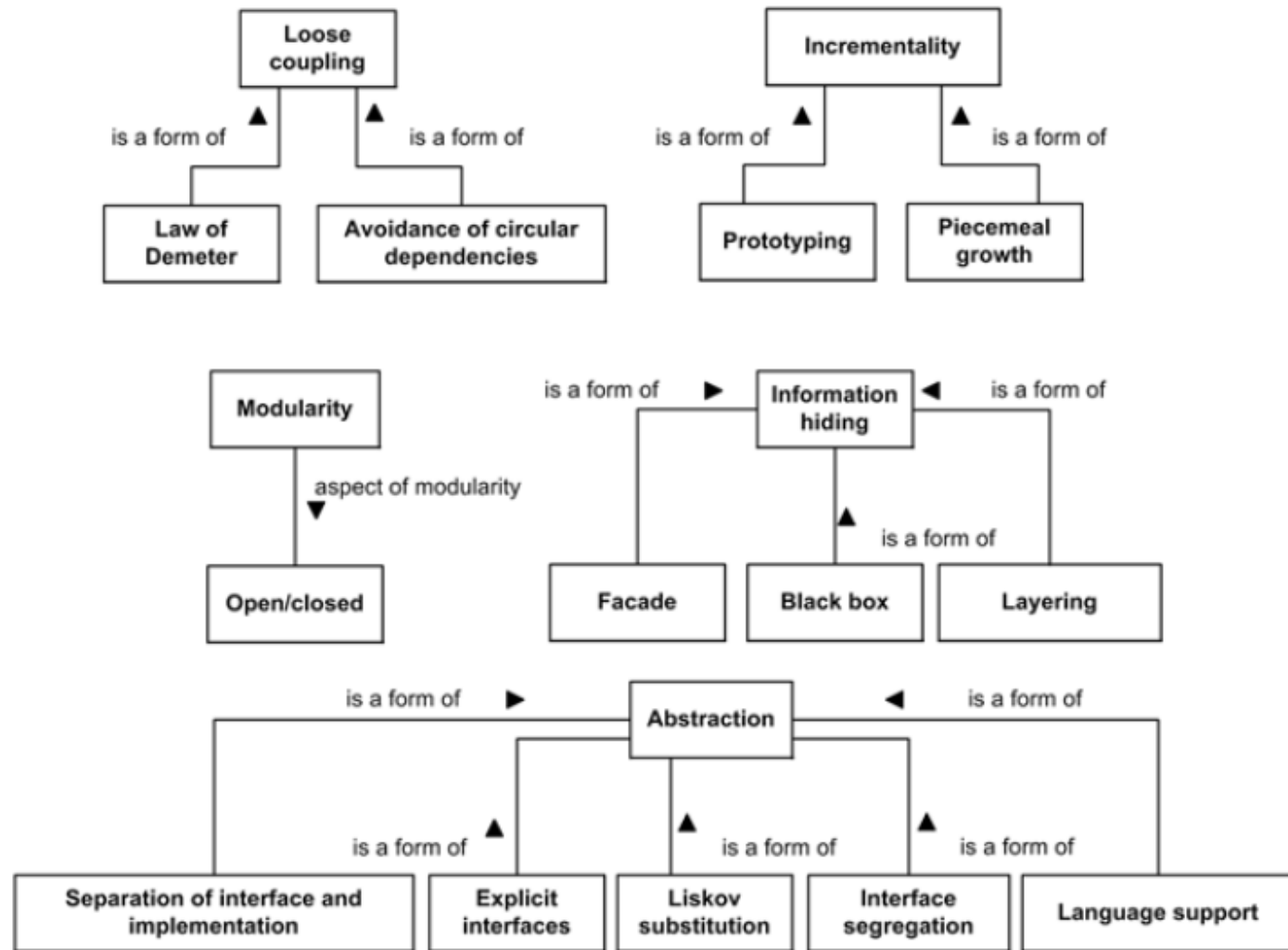
Basic concepts of the WITH WHAT dimension



Influence of architecture means



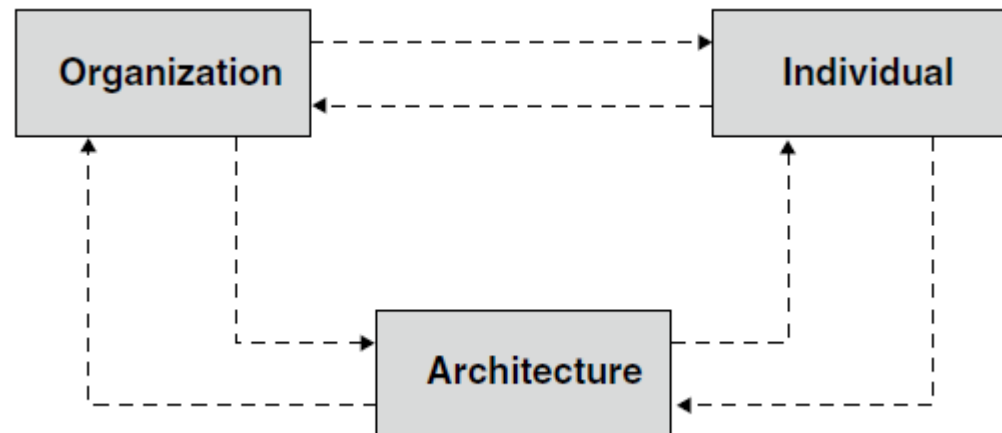
Overview of special variants of some architecture principles



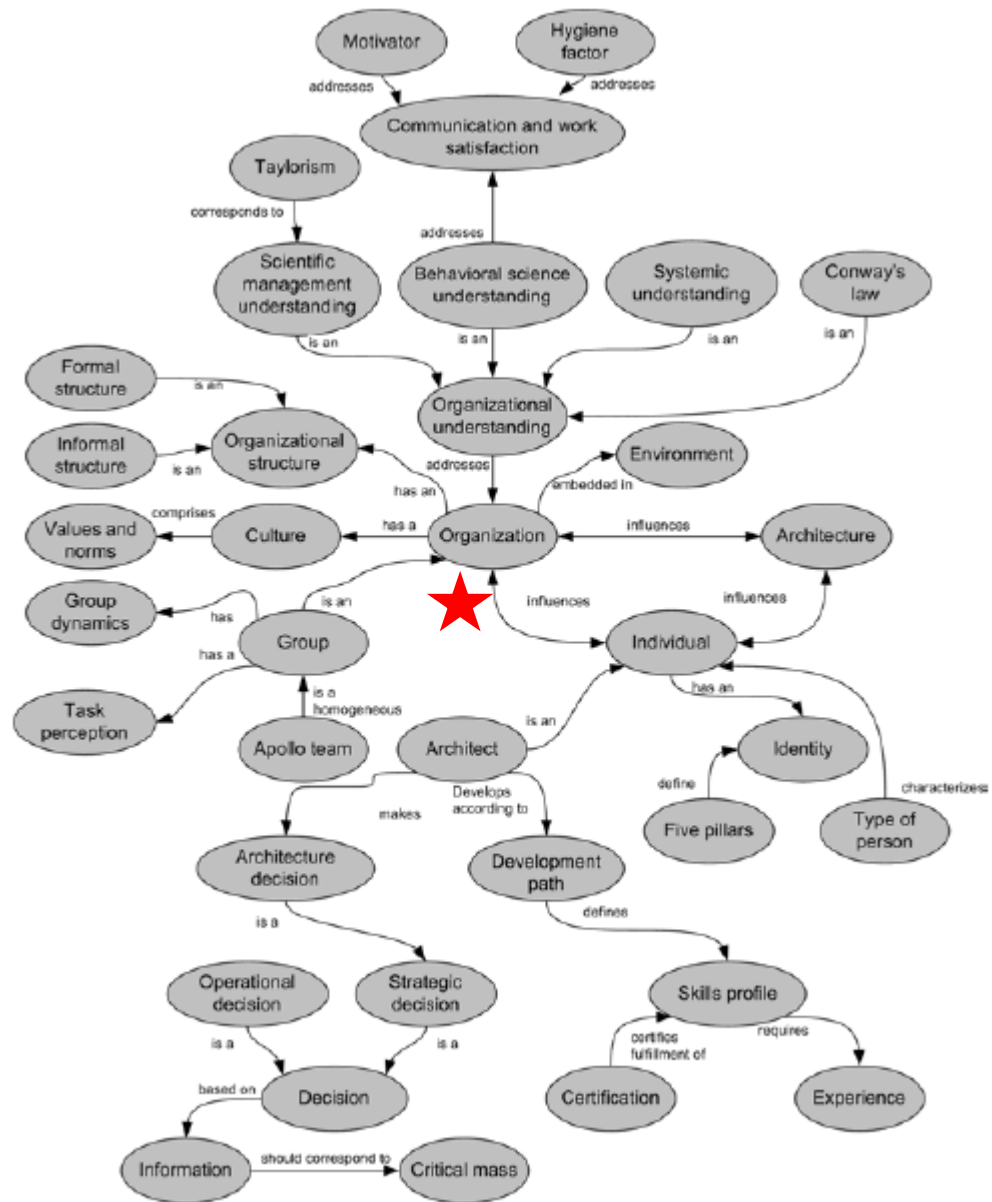
V. Organizations and Individuals (WHO)

Architectures are created by and for people in organizations

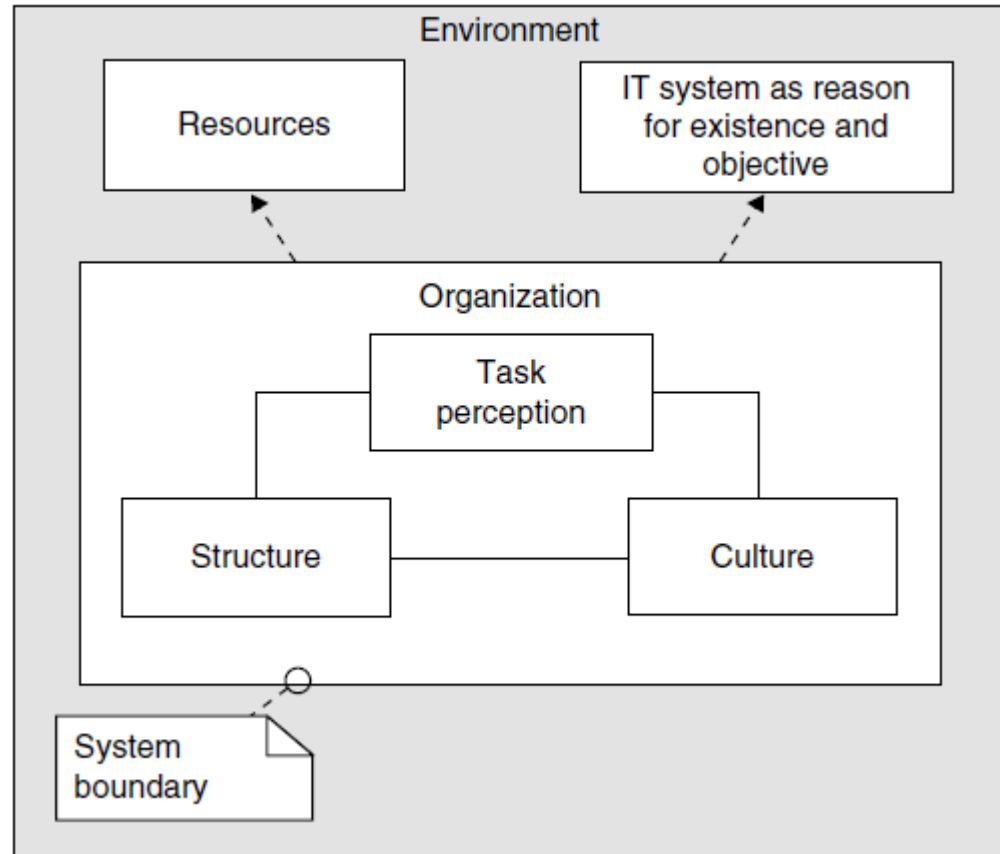
Influences of the organizational culture



Basic concepts of the WHO dimension



Systemic understanding





VI. Architecture Method (HOW)

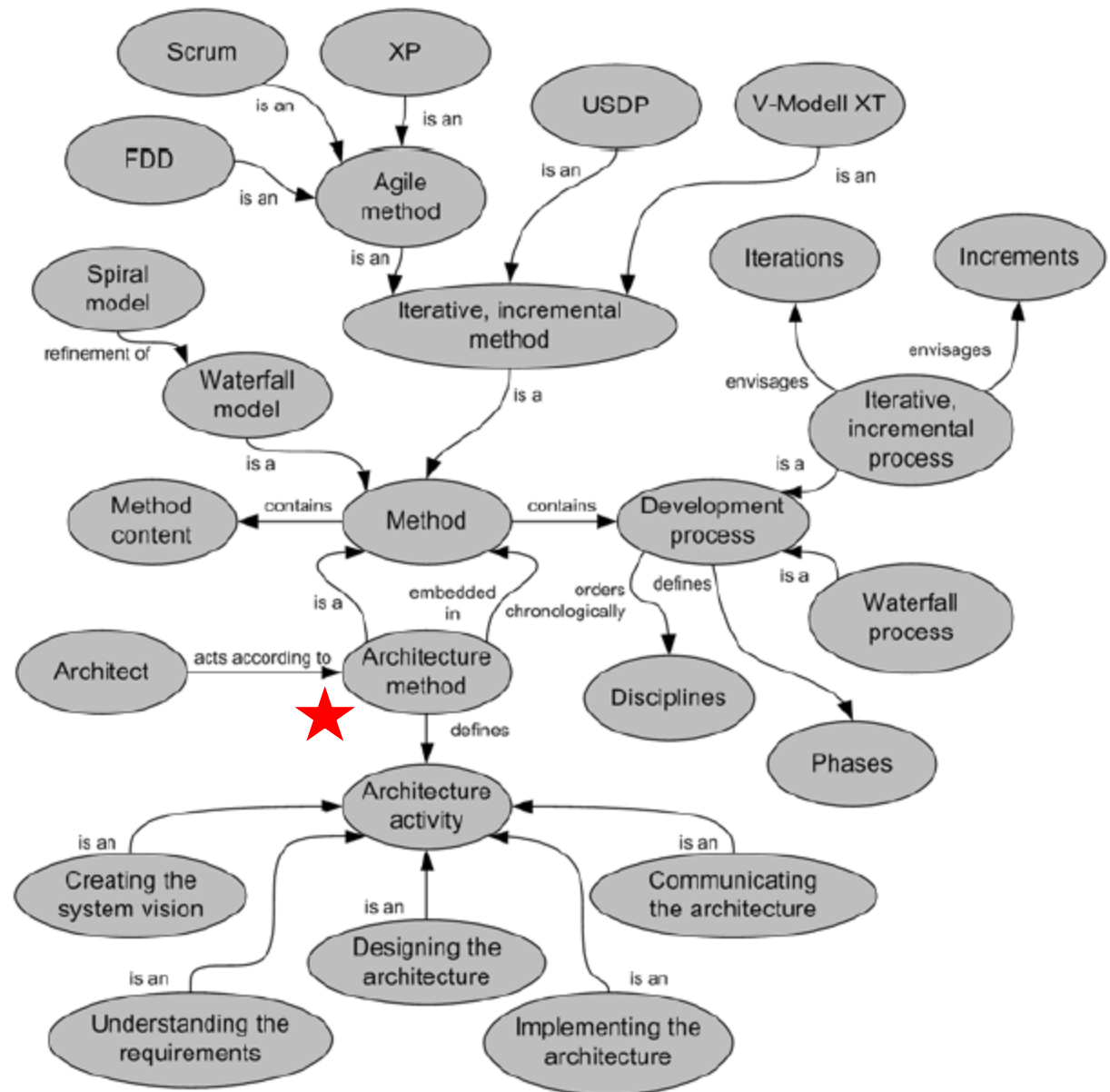
The architecture method contains the following **activities** that must be executed

- Creating the system vision
- Understanding the requirements
- Designing the architecture
- Implementing the architecture
- Communicating the architecture

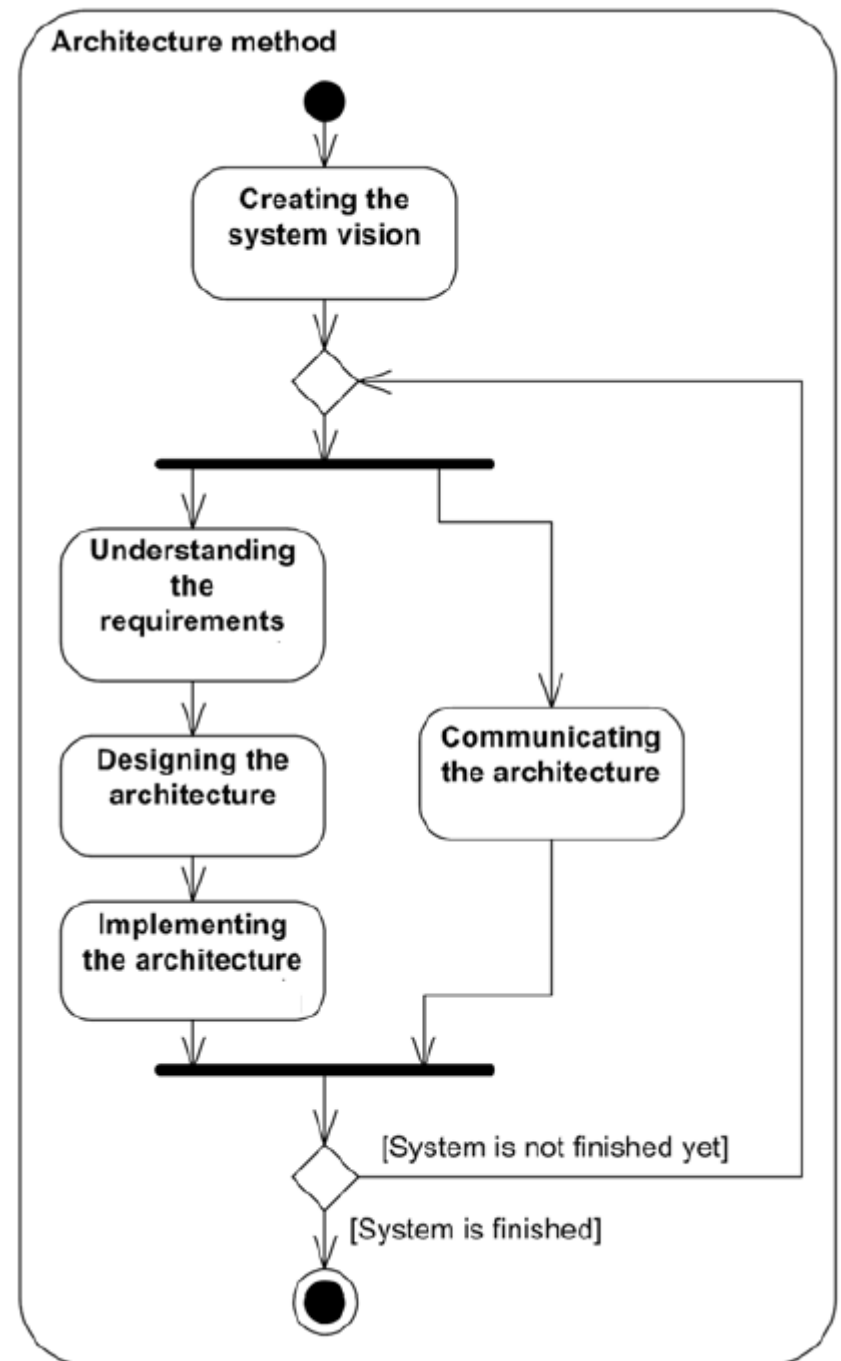
An architect needs a method and experience.

The typical activities of an architect.

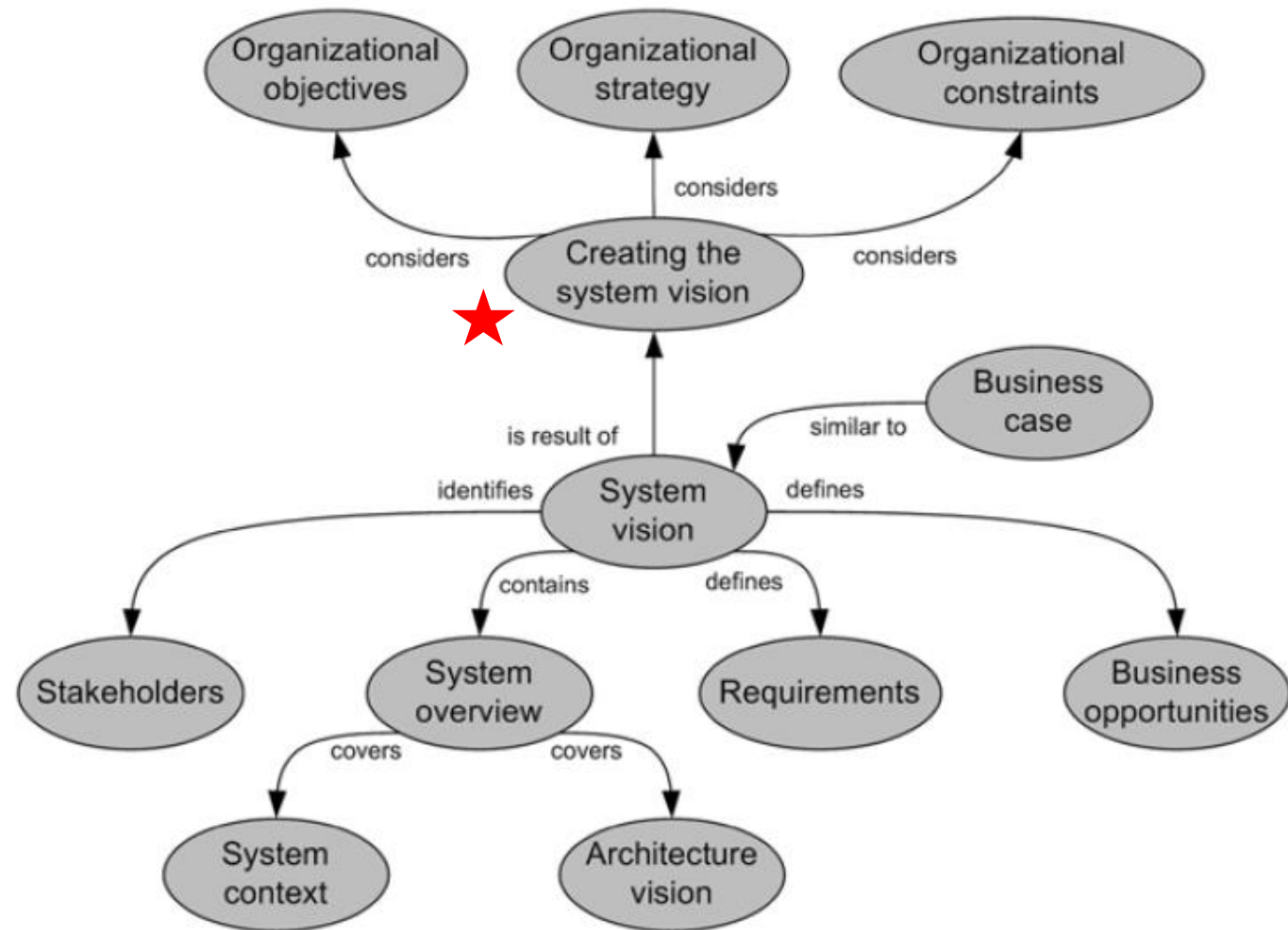
Basic concepts of the HOW dimension



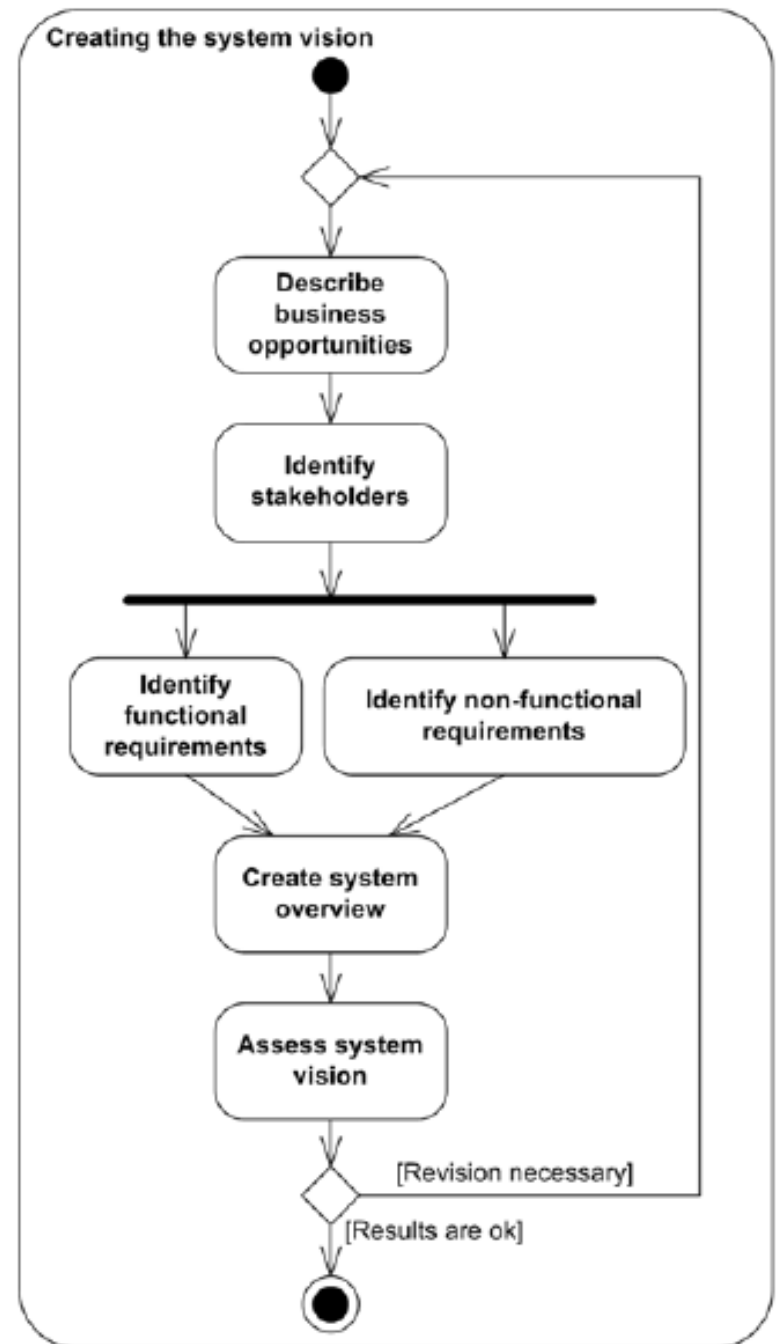
Overview of the architecture method



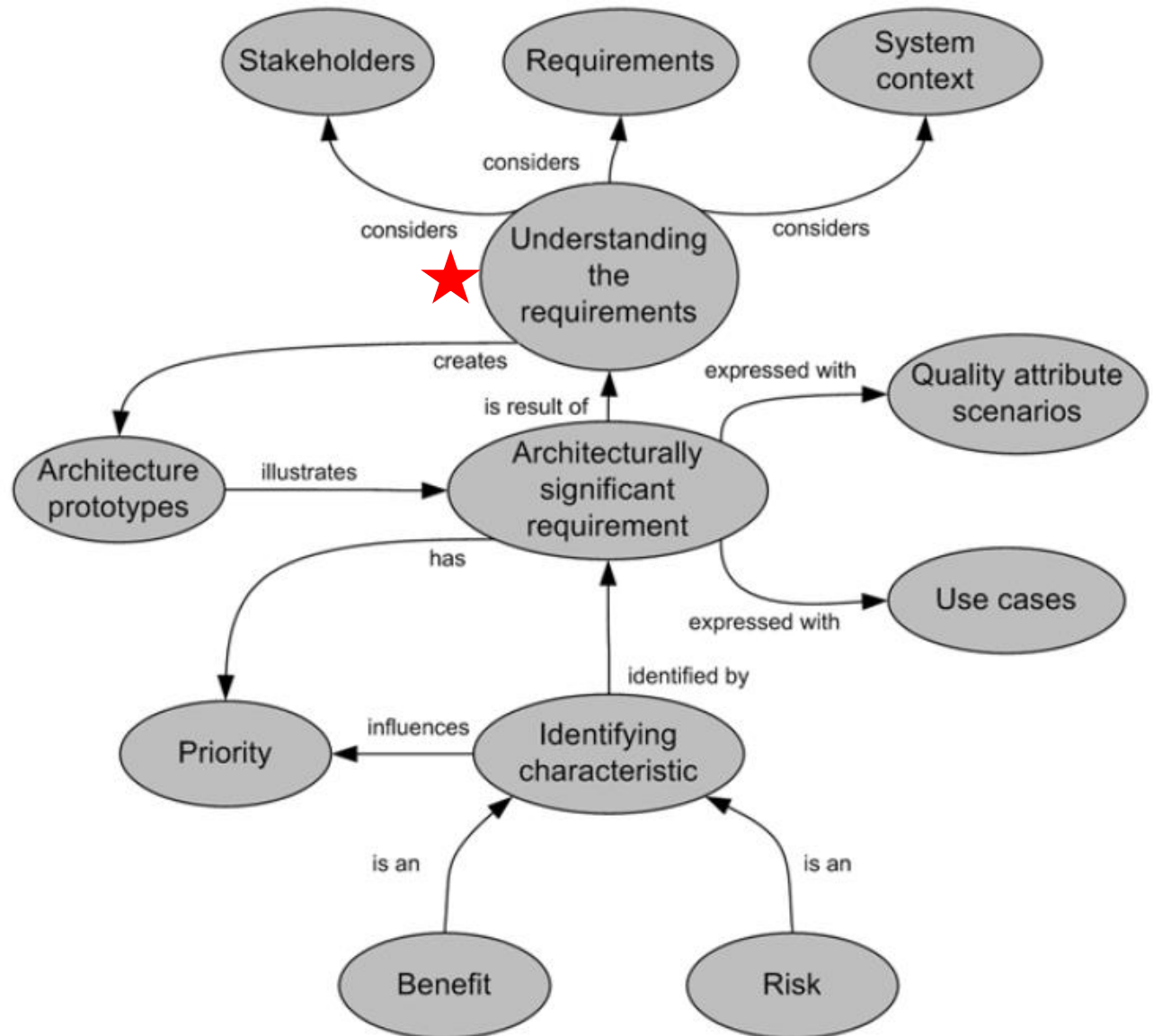
Creating the System Vision



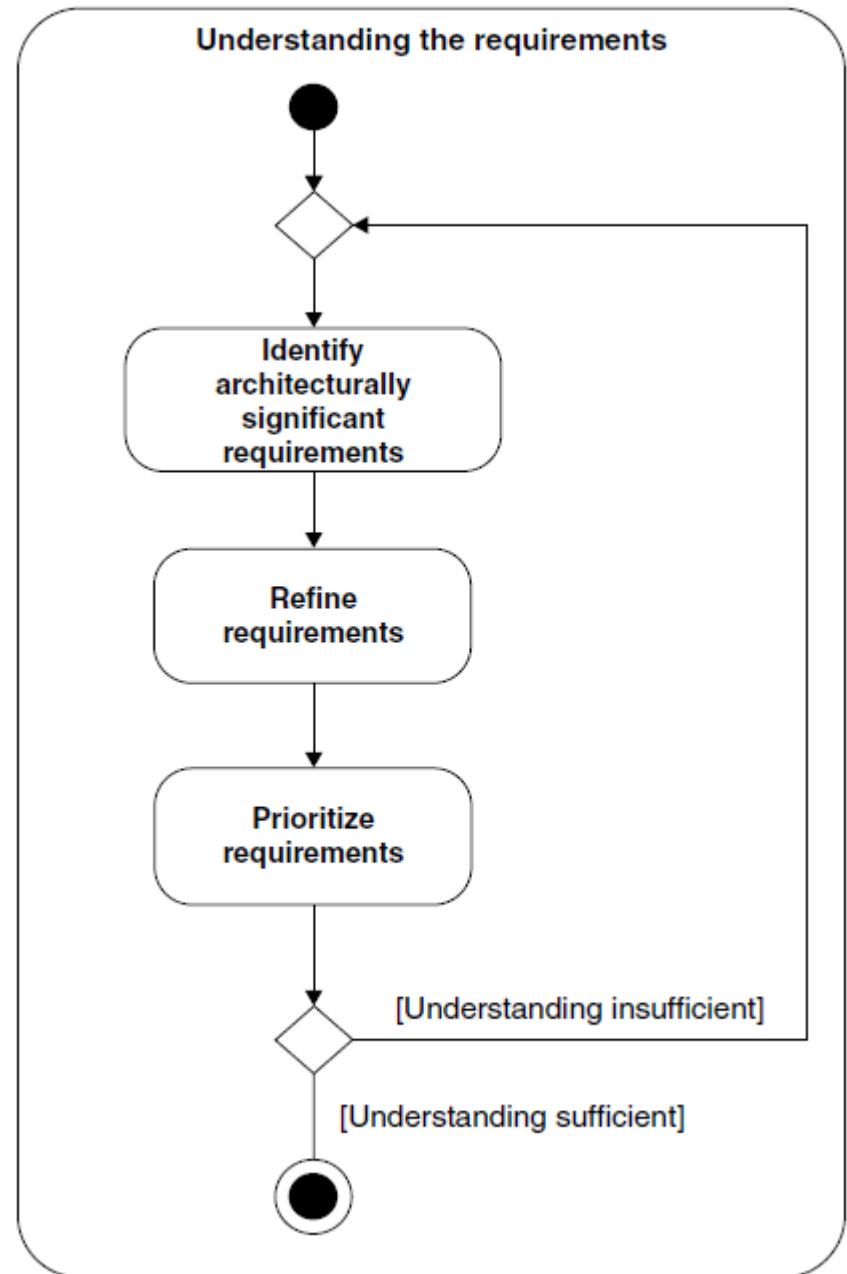
Creating the system vision



Understanding the Requirements



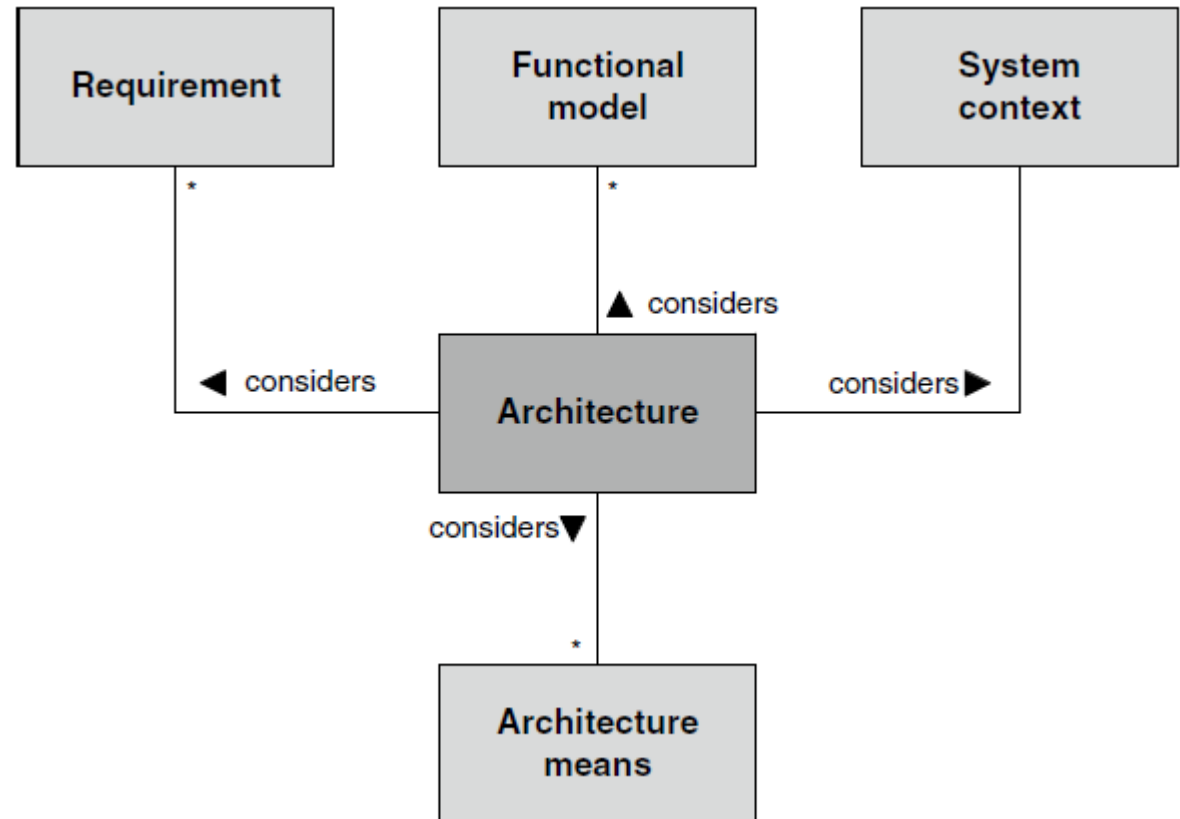
Understanding the requirements



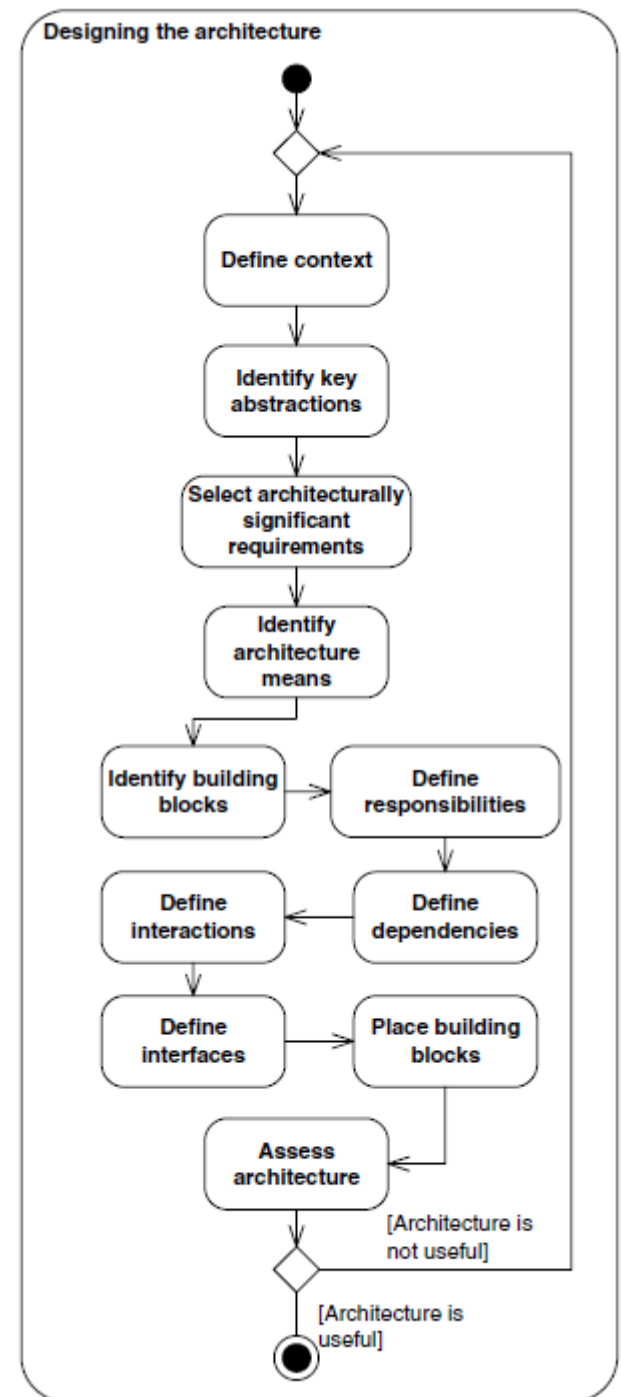
Designing the Architecture



Influencing factors on the architecture design



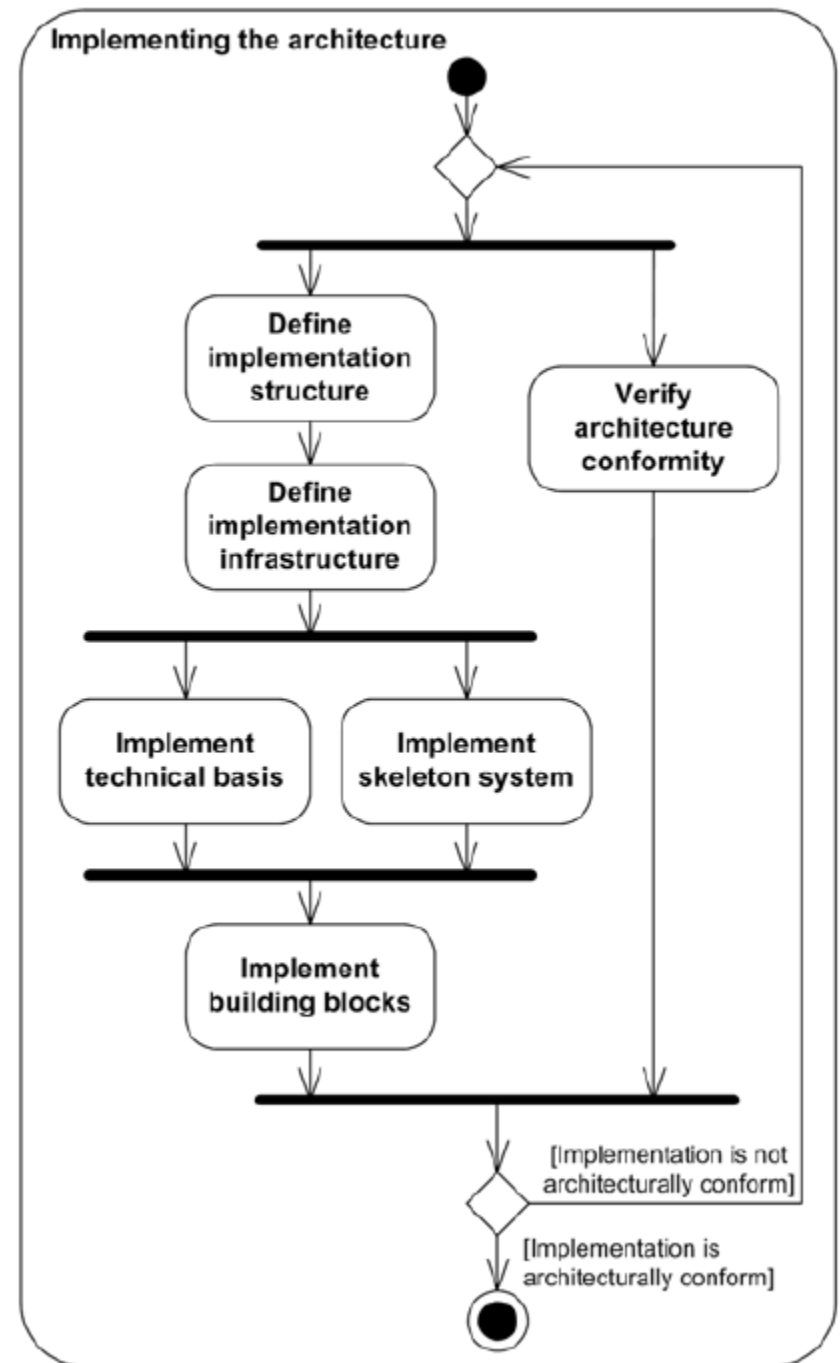
Designing the architecture



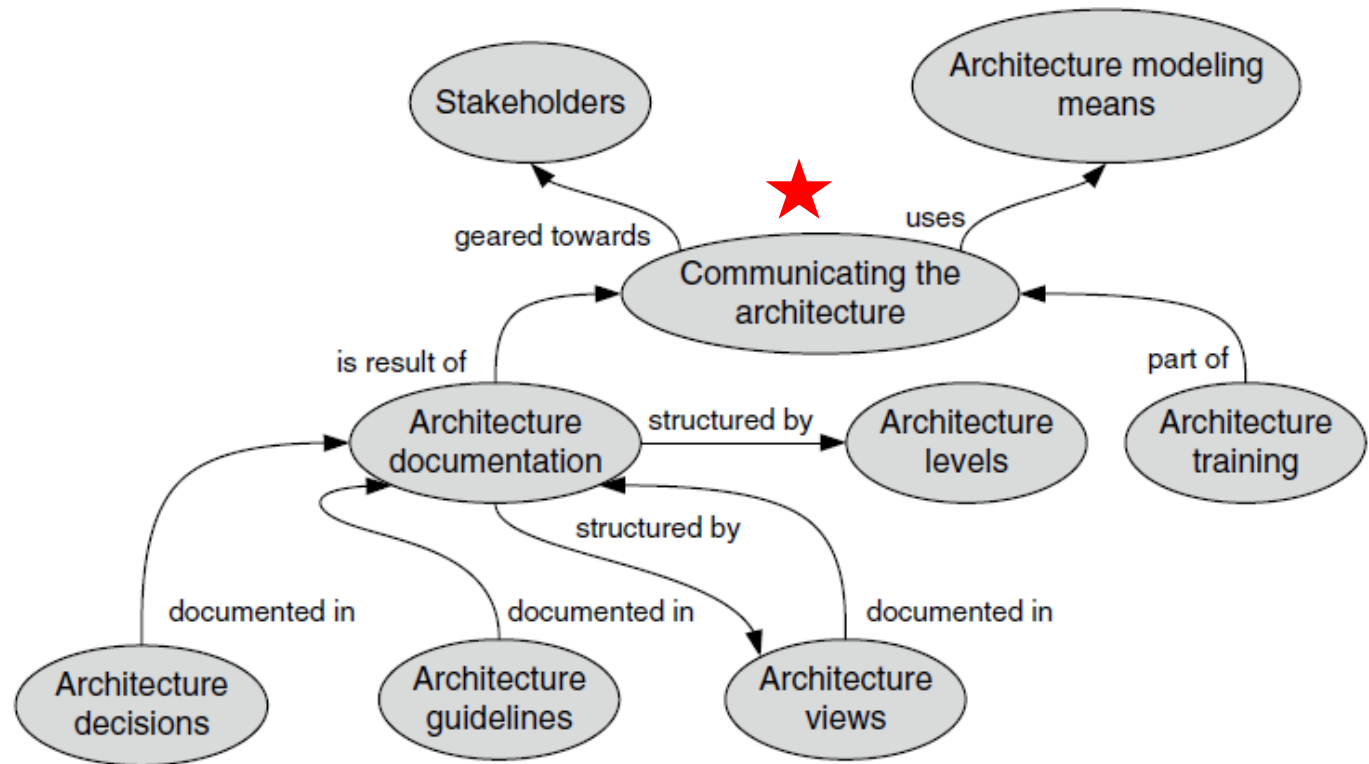
Implementing the Architecture



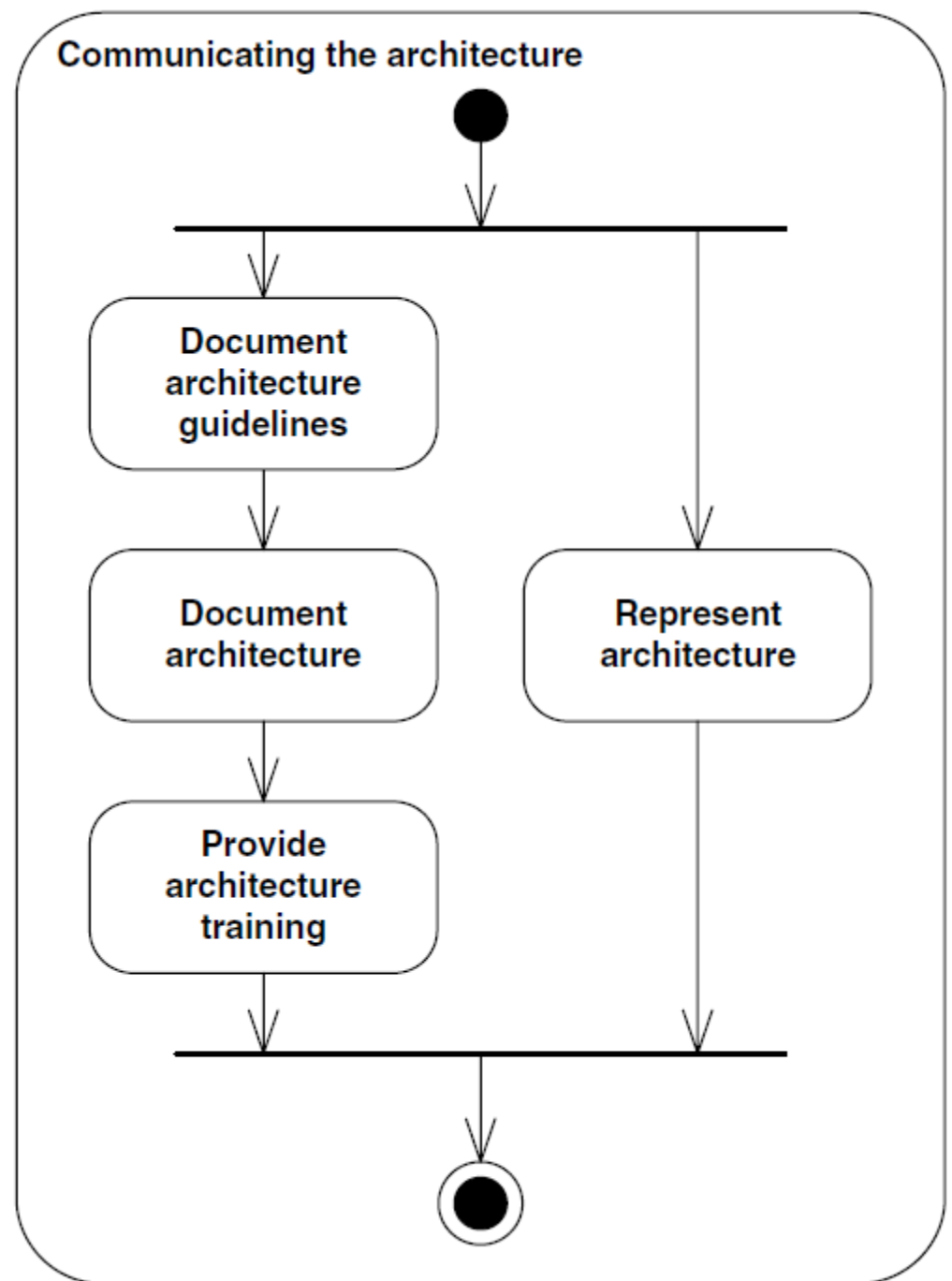
Implementing the architecture



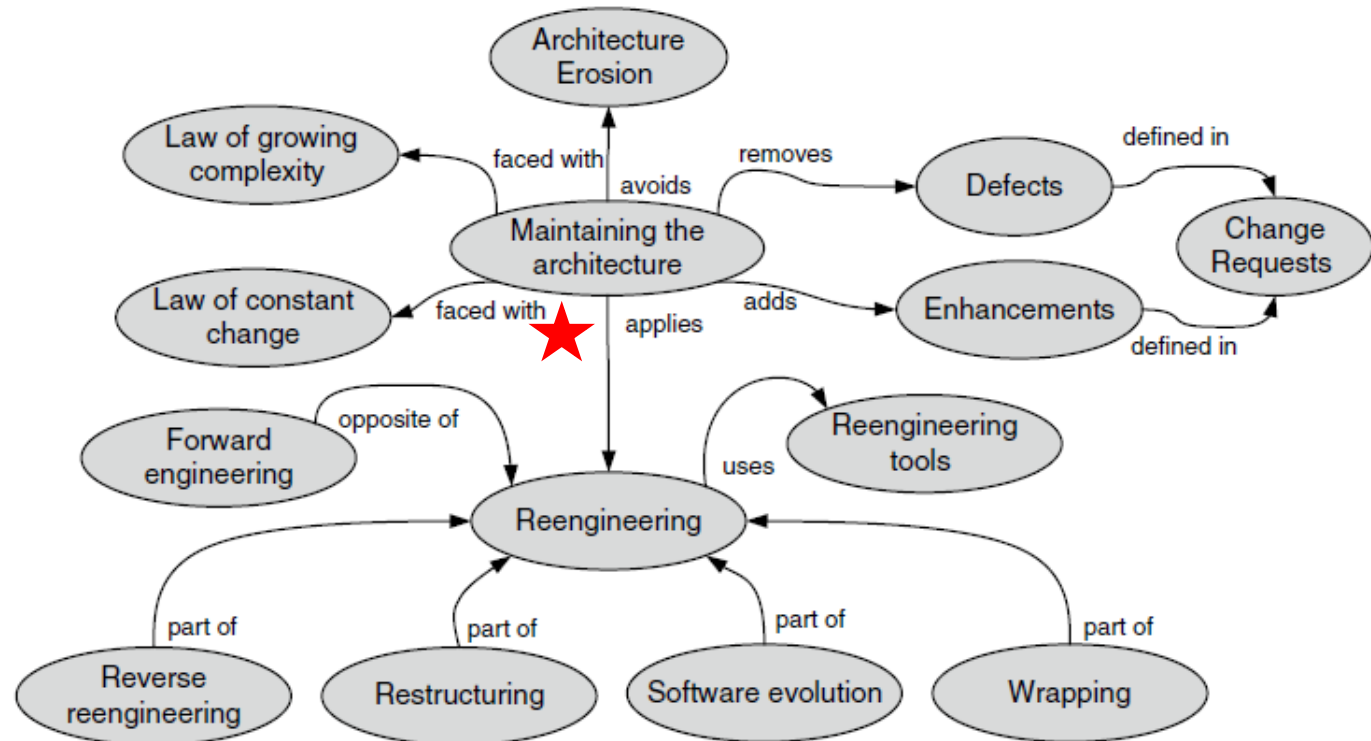
Communicating the Architecture



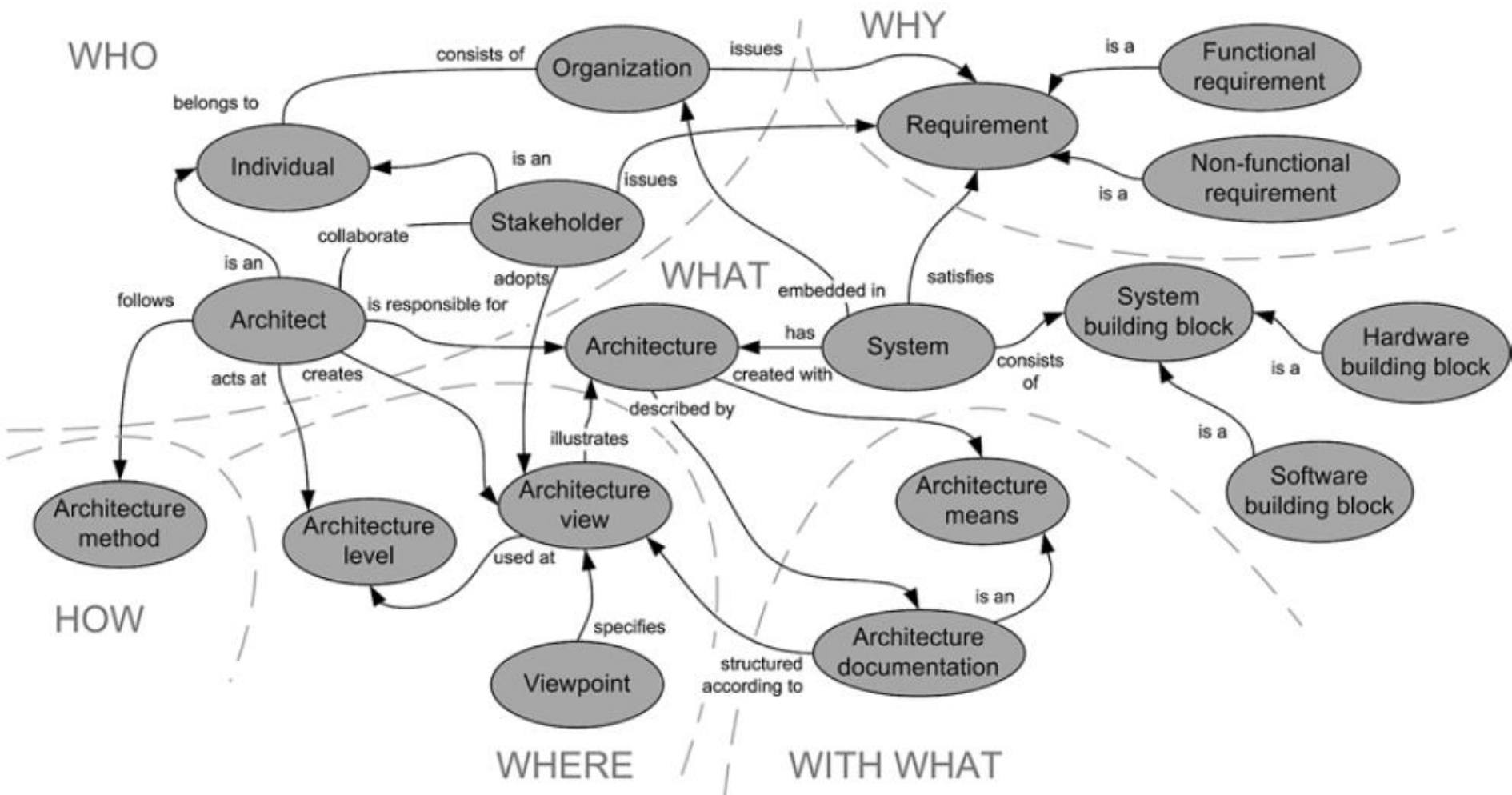
Communicating the architecture



Maintaining the Architecture



Software architecture domain



Any Questions?



✉ hvusynh@hcmiu.edu.vn