

SYS466

Analysis and Design

Lecture 1 - Software Architecture Design
School of Information and Communications Technology
Seneca College

Terminology

- analysis - decomposing something into its components

goal: conceptual model of solution

- design - create concrete solution from conceptual model

- architecture

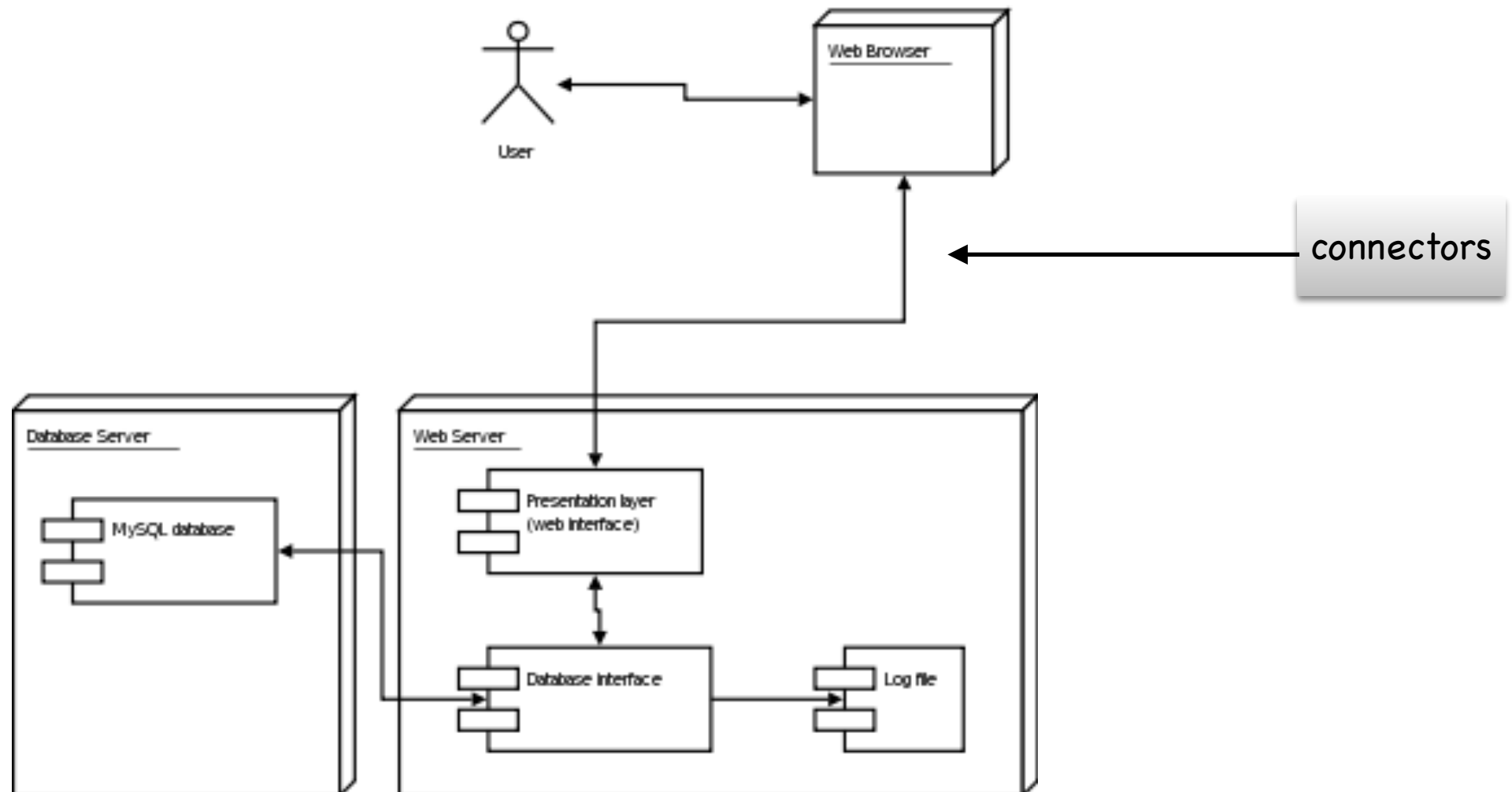
- organization of components and relationships

- how organization and its development are viewed

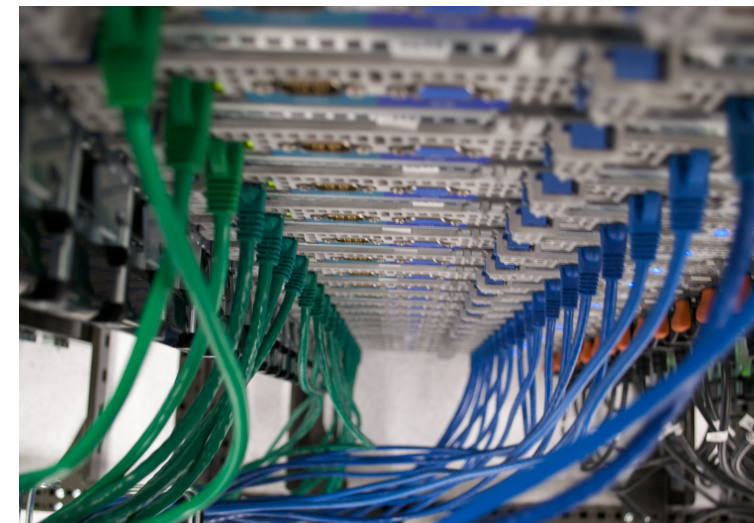
- properties should make it meet system requirements

Architecture Design

“..focuses on the decomposition of a system into components and the interaction between those components in order to satisfy functional and non-functional requirements..” (Albin 2003)



System Requirements



primarily focuses on
correctness

focuses on how
correctness is
achieved

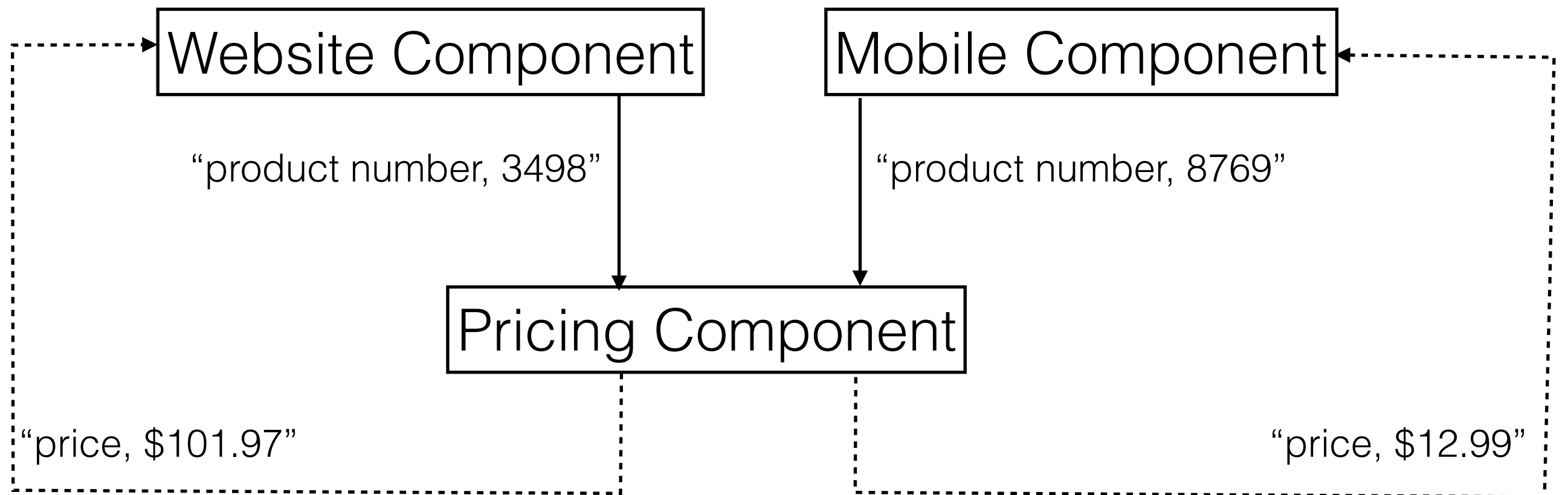
Functional

Non-functional

Software Design

decisions are binding
contracts between
components





Design-by-contract example

"if **you give me** a valid product number...
I **will give you** the lowest available price!"

IT Architecture

- more than software (hardware, network topology, OS)
- influences software design decisions

Architecture Design Process

- Understand the problem (use case analysis)
- Identify design elements and their relationships
- Evaluate the design
- Transform/refine the design

Architecture View

- a representation of a system
 - 📌 from a particular perspective (behavioural, structural, dynamic)
 - 📌 for a stakeholder
 - 📌 at a certain point in the development process
 - 📌 of a particular subsystem

Architectural Patterns

Patterns

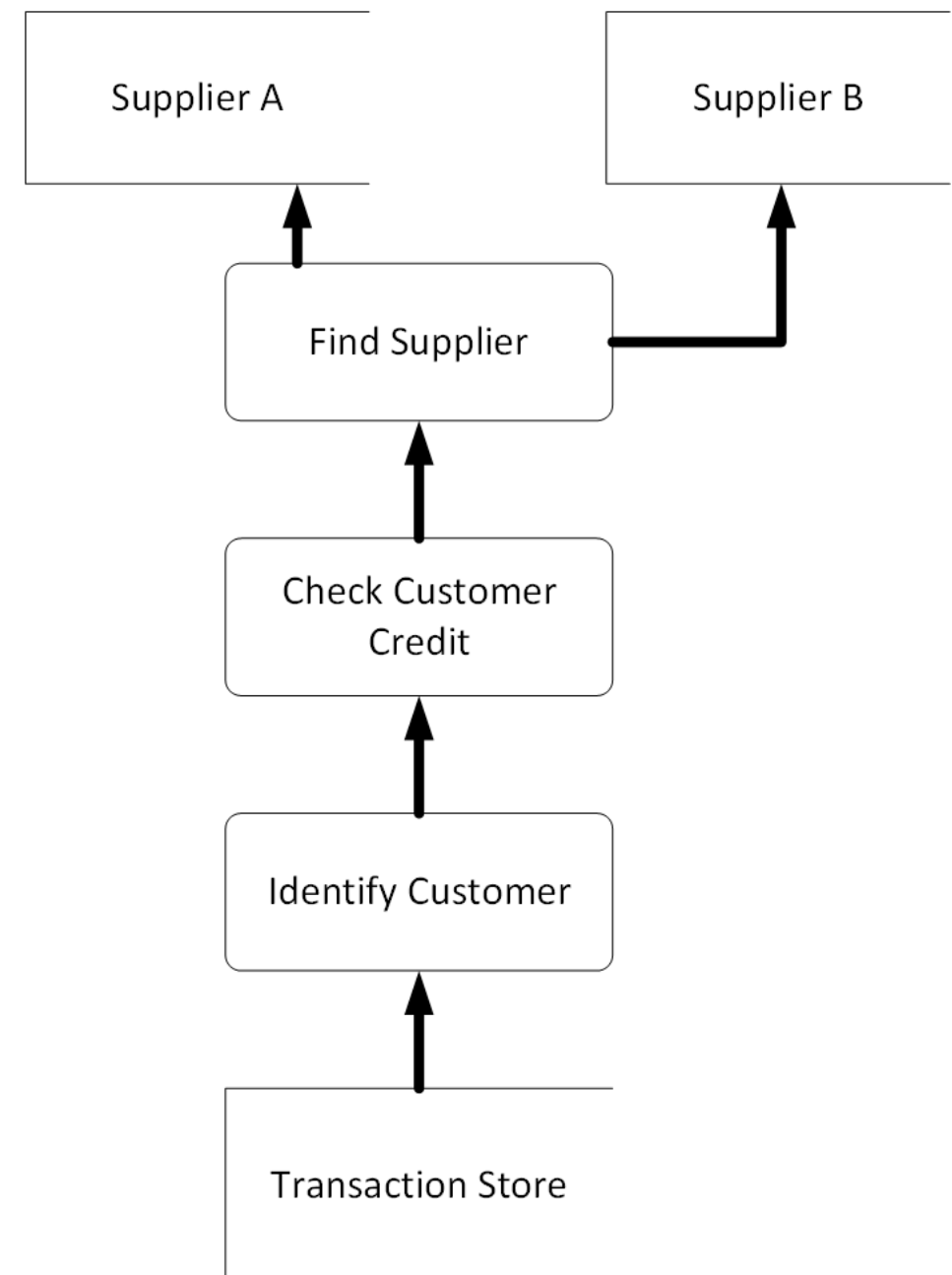
- proven approach to solving a problem
- observed in multiple solutions
- provides common vocabulary for experts

Architectural Patterns

- architecture is pattern-driven, rarely do architects design from scratch
- an architecture of a system is created by combining and customizing known patterns

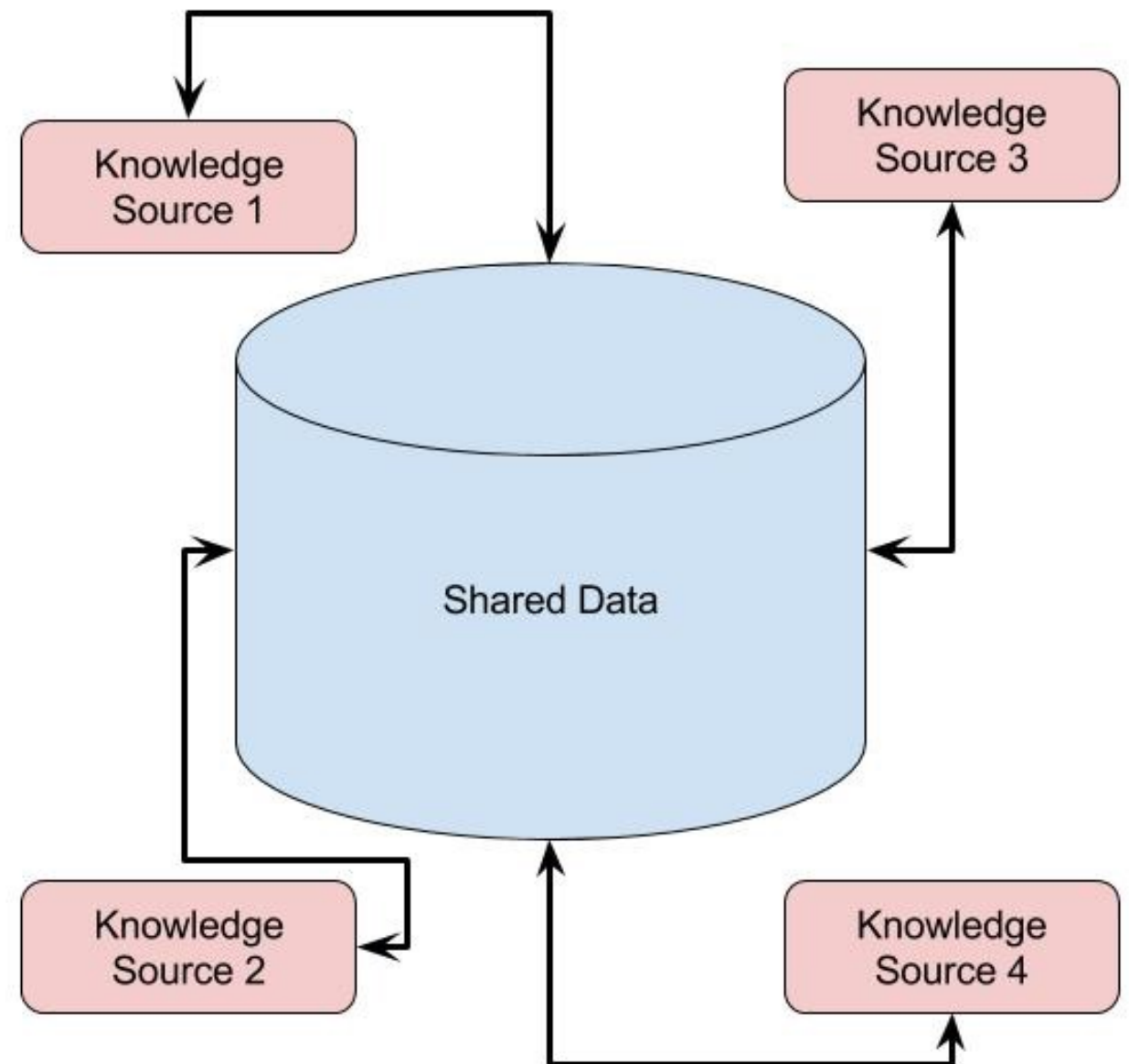
Pipe and Filter Pattern

- components have a set of inputs and outputs
- input is incrementally modified
- appropriate for applications where data goes through a series of transformations



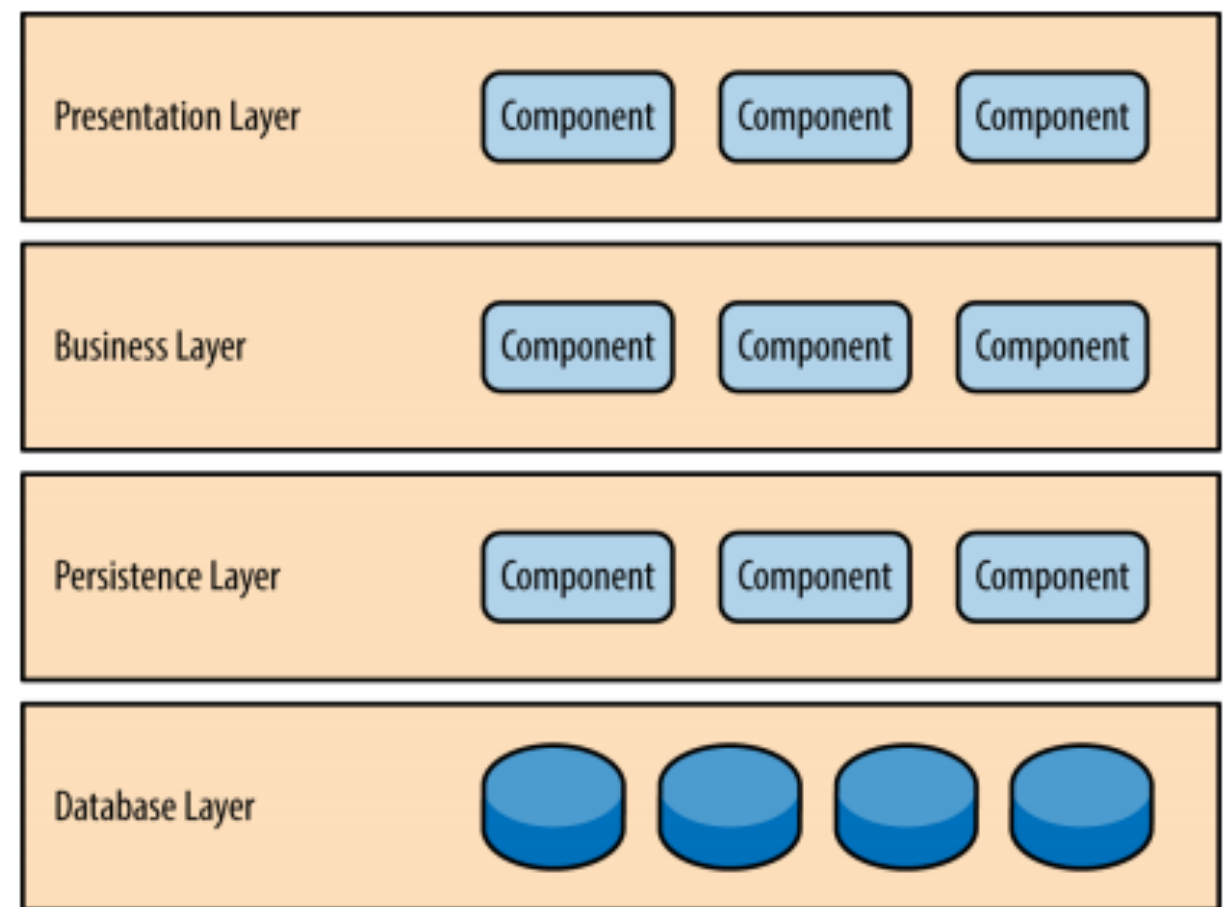
Repository Pattern

- central data structure (current state)
- collection of independent components with operate on store



Layering Pattern

- collect components that are highly dependent on each other into logical groups
- groups created should ideally not be dependent on other components
- horizontal layers are organized such that lower levels are not dependent on higher ones



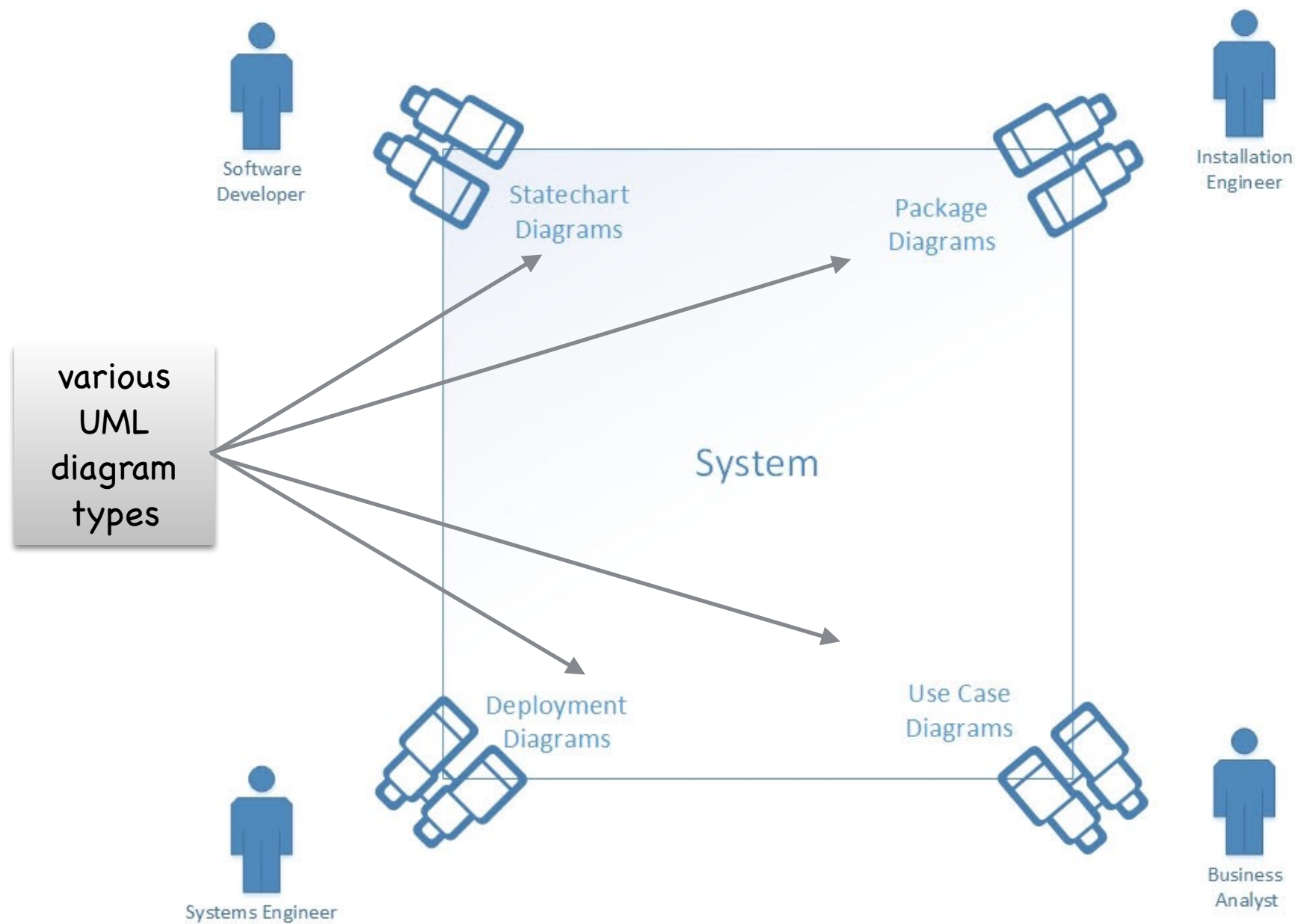
UML

- family of graphical notations used to describe and design software systems
- used to document multiple system views



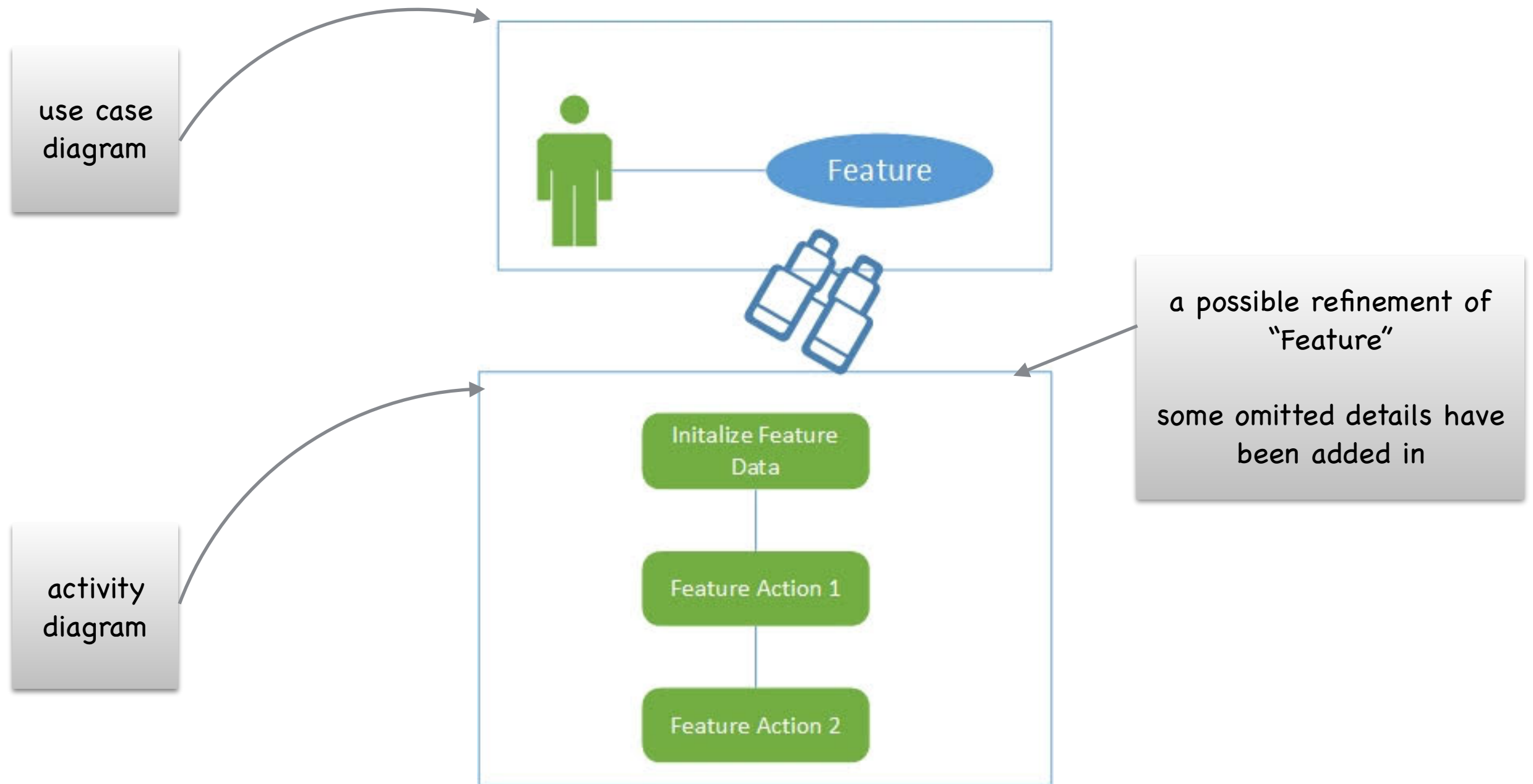
UML is ...

- a notation ...
- a meta-model



UML

...provides different views of the system for different stakeholders...



UML

...provides multiple levels of abstraction

UML Uses

- sketches, helps communicate some aspects of the system
- blueprints, complete description of design, enough detail for developer to implement
- programming language, compiled to executable code

References

- Object-Oriented Systems Analysis and Design (Ashrafi, 2009) Chapter 16
- The Art of Software Architecture, (S Albin 2003), Chapter 3
- Introduction to Software Architecture, (Garlan, Shaw, 1994)
- UML Distilled, 3rd Edition (Fowler, 2004)