Chapter 10
Architectural Design

Software Engineering: A Practitioner's Approach, 6th edition
by Roger S. Pressman

概述

- 人们不能在没有图纸的情况下建房子, 同样也不能通过勾画出房子的管道布局而开始绘制房子的蓝图. 在开始考虑细节之前, 需要关注整体视图---房子本身. 这就是体系结构设计需要做的事情---它为你提供整体的视图并保证得到正确的理解.

Software Architecture

- The software architecture of a program or computing system **is the structure or structures of the system**, which comprise the software components, the externally visible properties of those components, and the relationships among them.

— Bass. et al.

Why Architecture?

- Architecture is a representation of a system that enables the software engineer to:
  - analyze the effectiveness of the design in meeting its stated requirements,
  - consider architectural alternatives at a stage when making design changes is still relatively easy, and
  - reduce the risks associated with the construction of the software.

Data Design

- Architectural level → Database design
  - data mining
  - data warehousing
- Component level → Data structure design

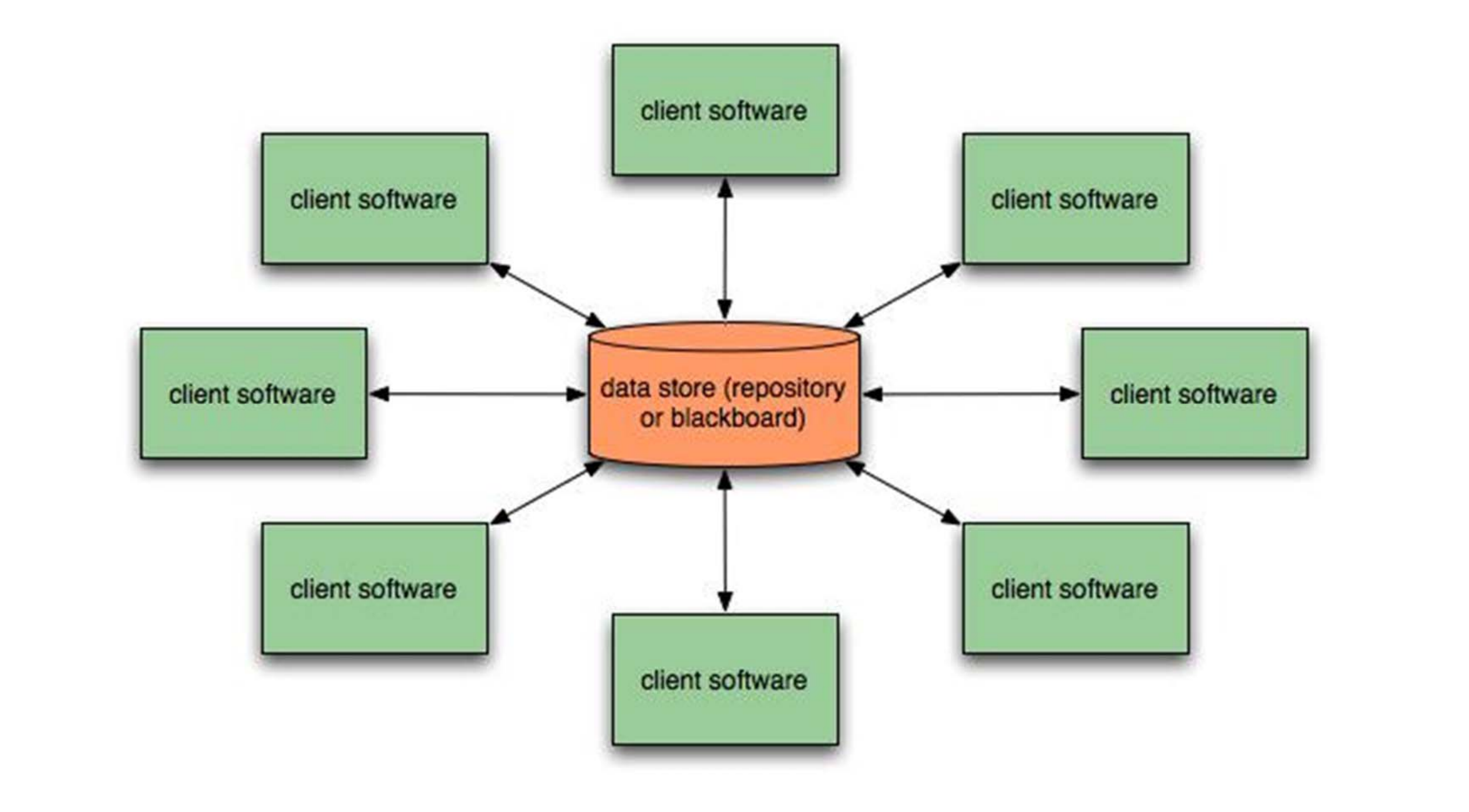
Architectural Styles

- Each style describes a system category that encompasses:
  - a set of components (e.g., a database, computational modules) that perform a function required by a system,
  - a set of connectors that enable “communication, coordination, and cooperation” among components,
  - constraints that define how components can be integrated to form the system, and
  - semantic models that enable a designer to understand the overall properties of a system.

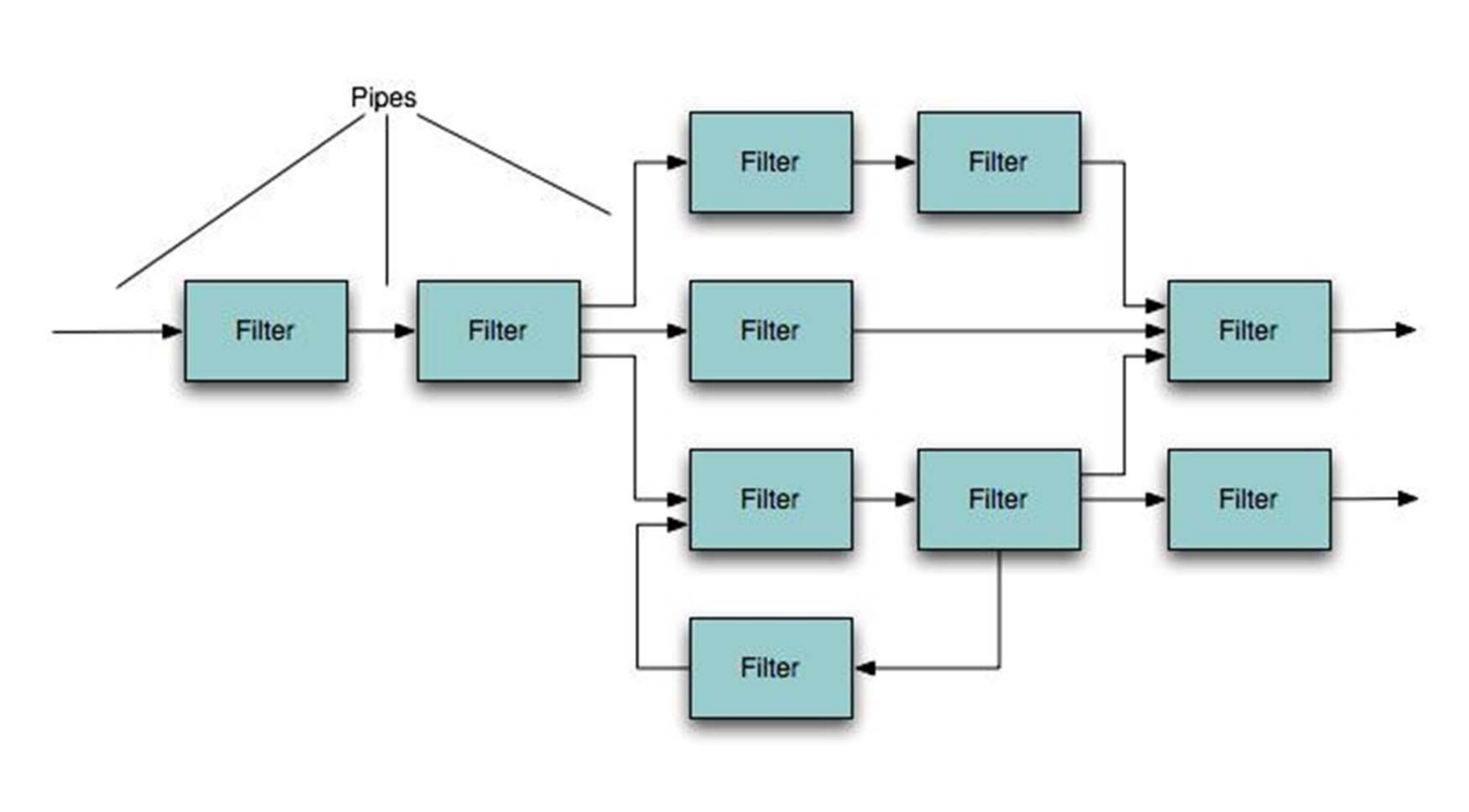
Specific Styles

- ◆ Data-centered architecture
- ◆ Data flow architecture
- ◆ Call and return architecture
- ◆ Object-oriented architecture
- ◆ Layered architecture

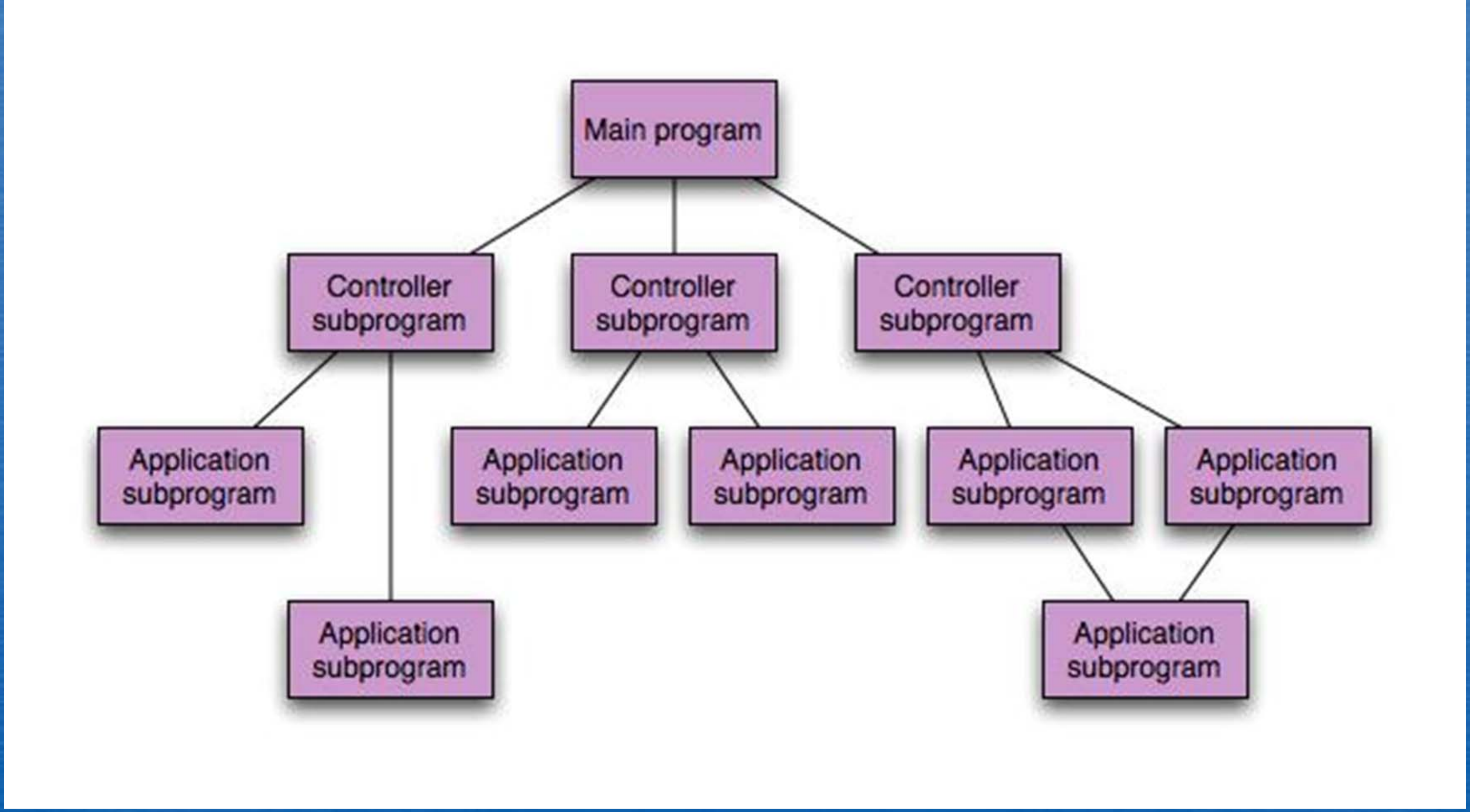
Data-Centered Architecture



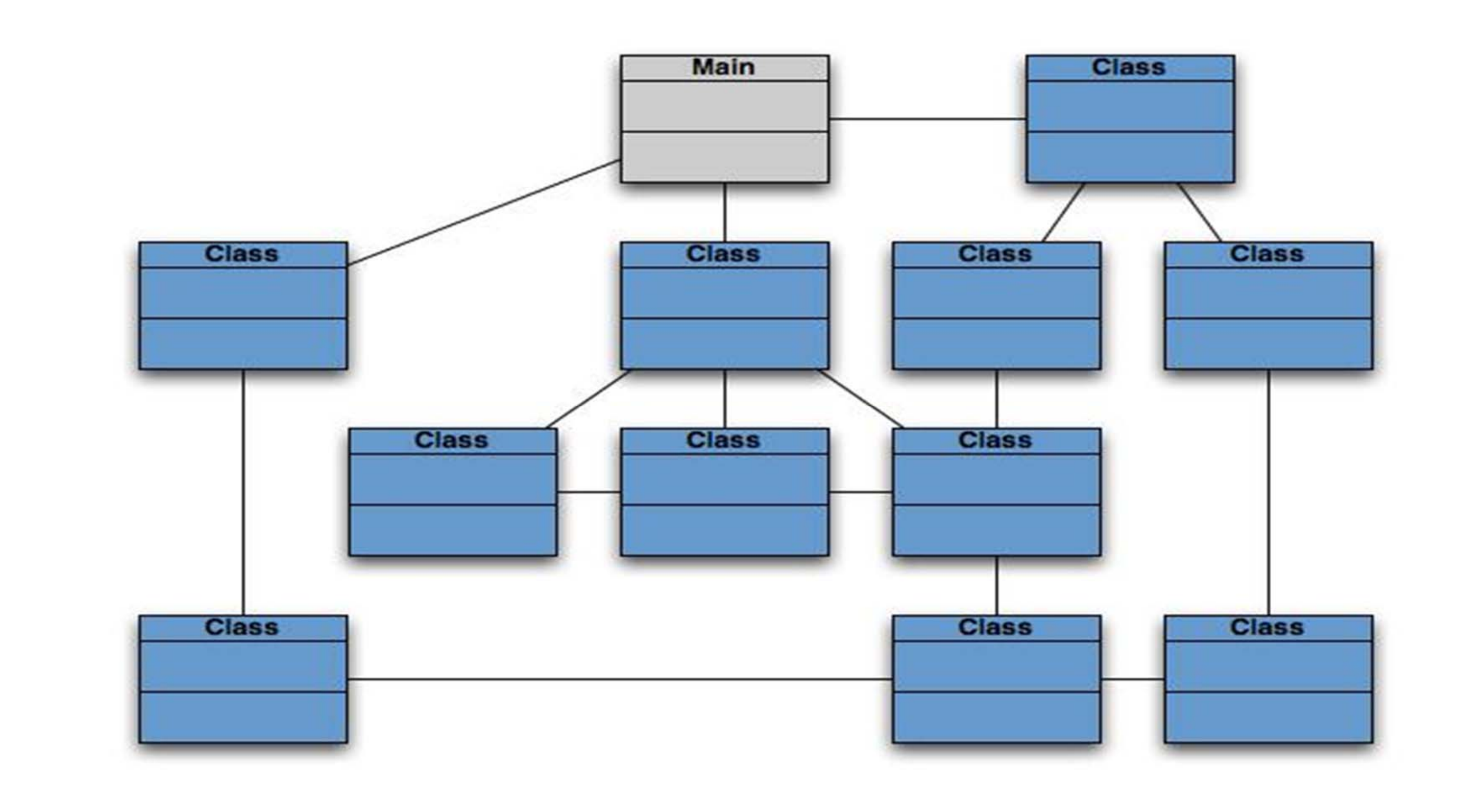
Data-Flow Architecture



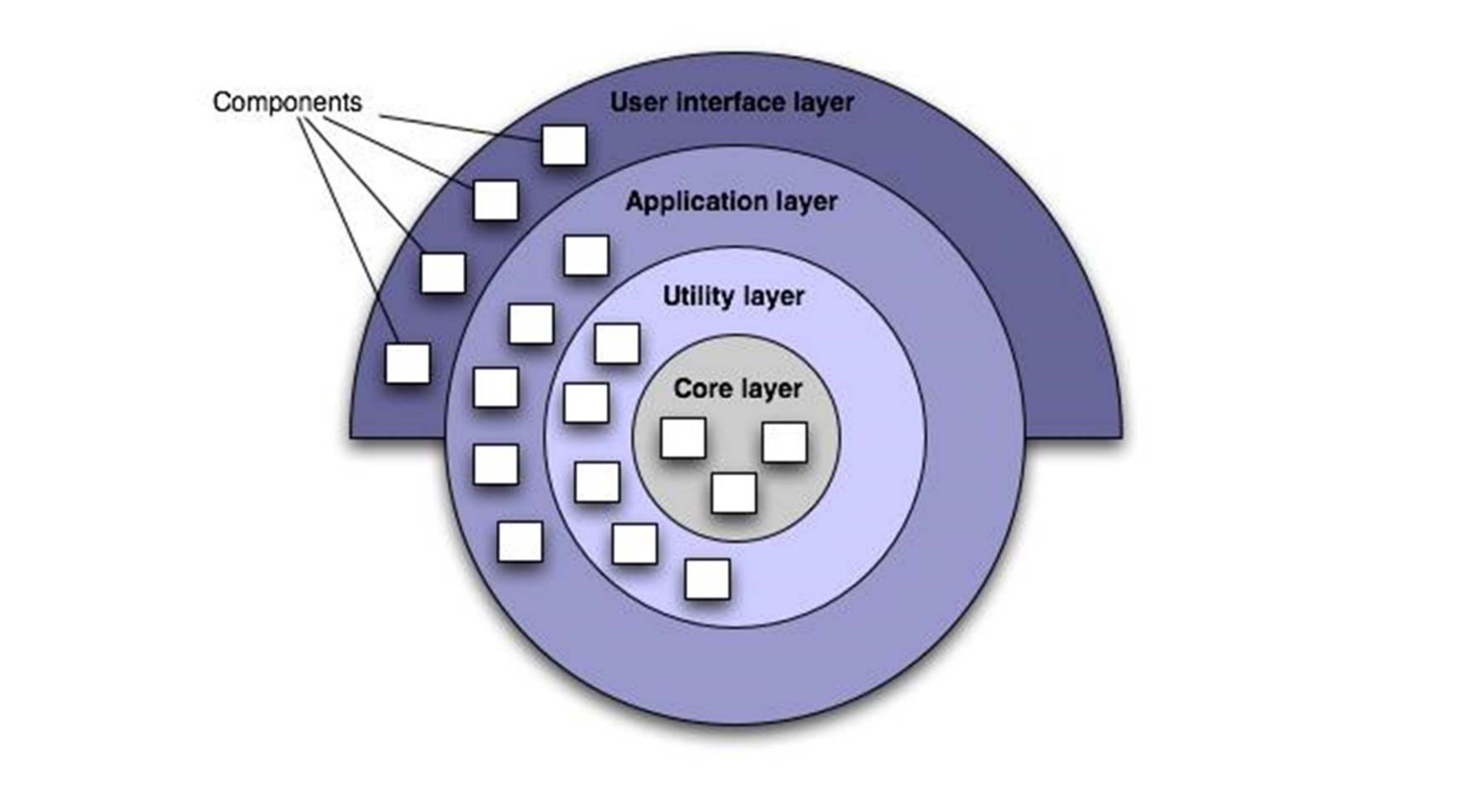
Call and Return Architecture



Object-Oriented Architecture



Layered Architecture



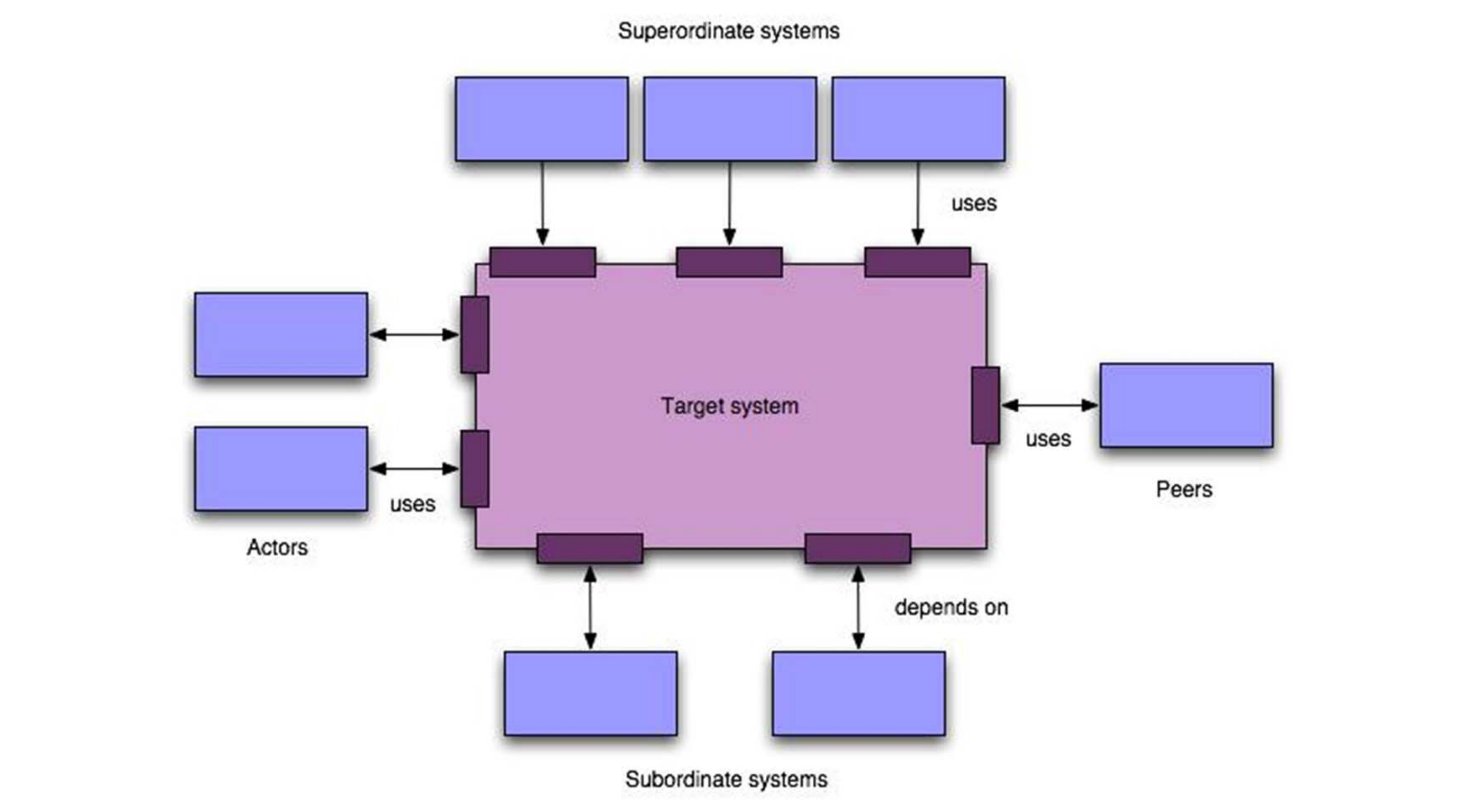
Architectural Patterns

- Concurrency
  - operating system process management
  - task scheduler
- Persistence
  - database management system
  - application level persistence
- Distribution
  - broker

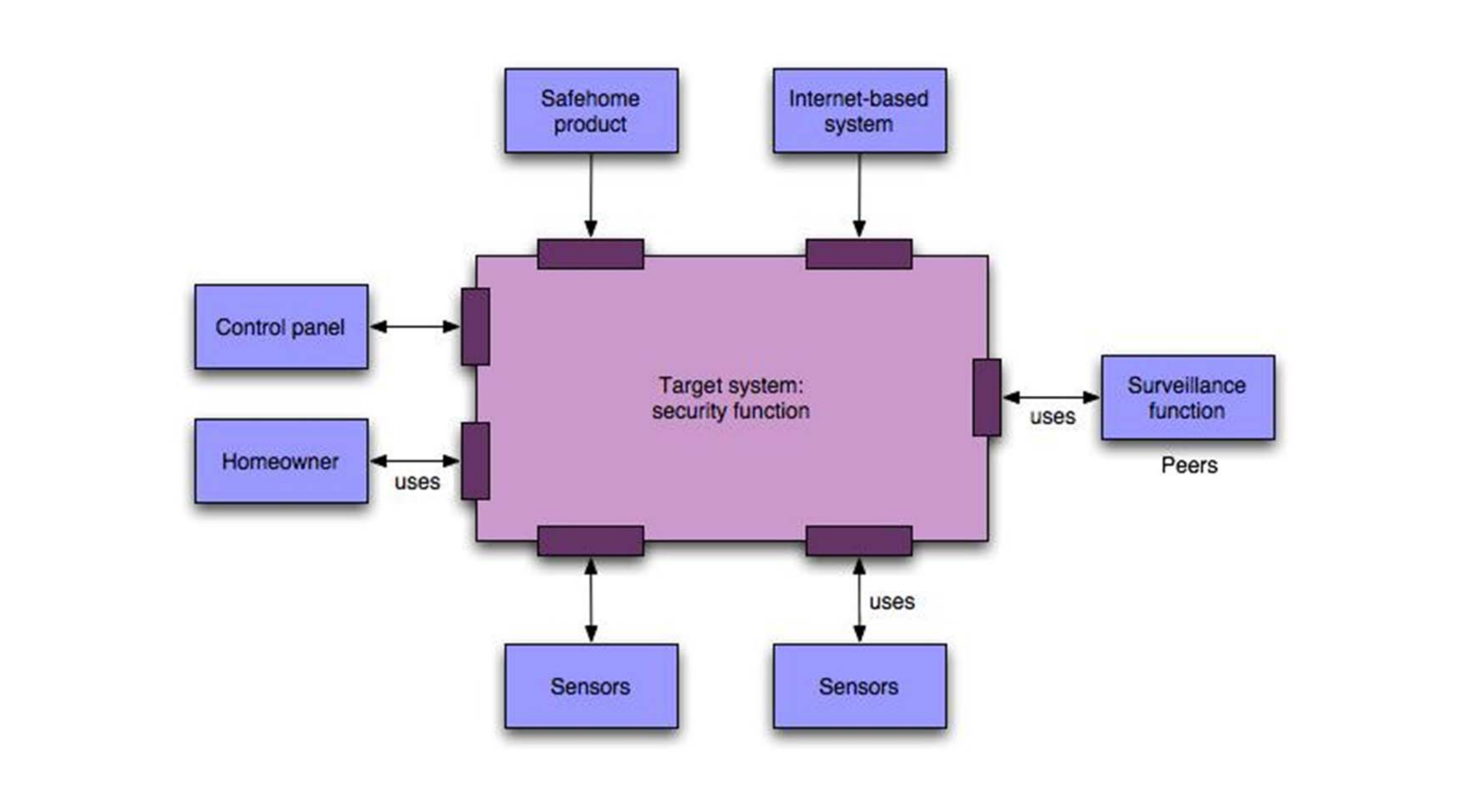
Architectural Design

- Architectural context diagrams model how software interacts with external entities
- Archetypes are classes or patterns that represent an abstraction critical to the system
- Architectural components are derived from the application domain, the infrastructure, and the interface.

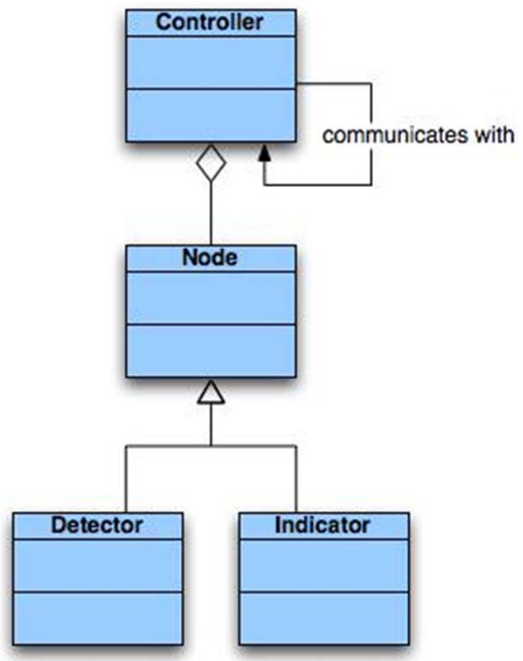
Arch. Context Diagram



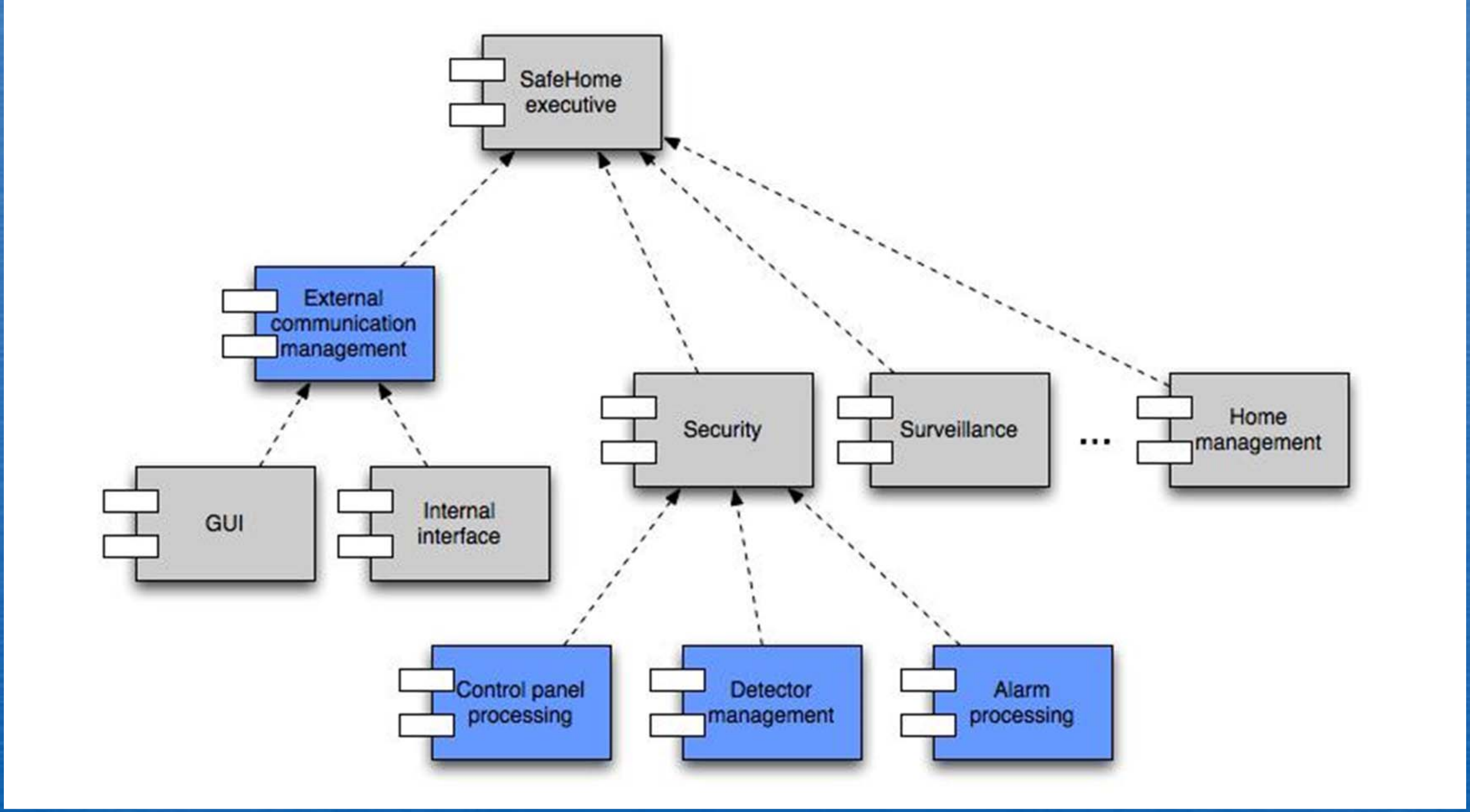
SafeHome ACD



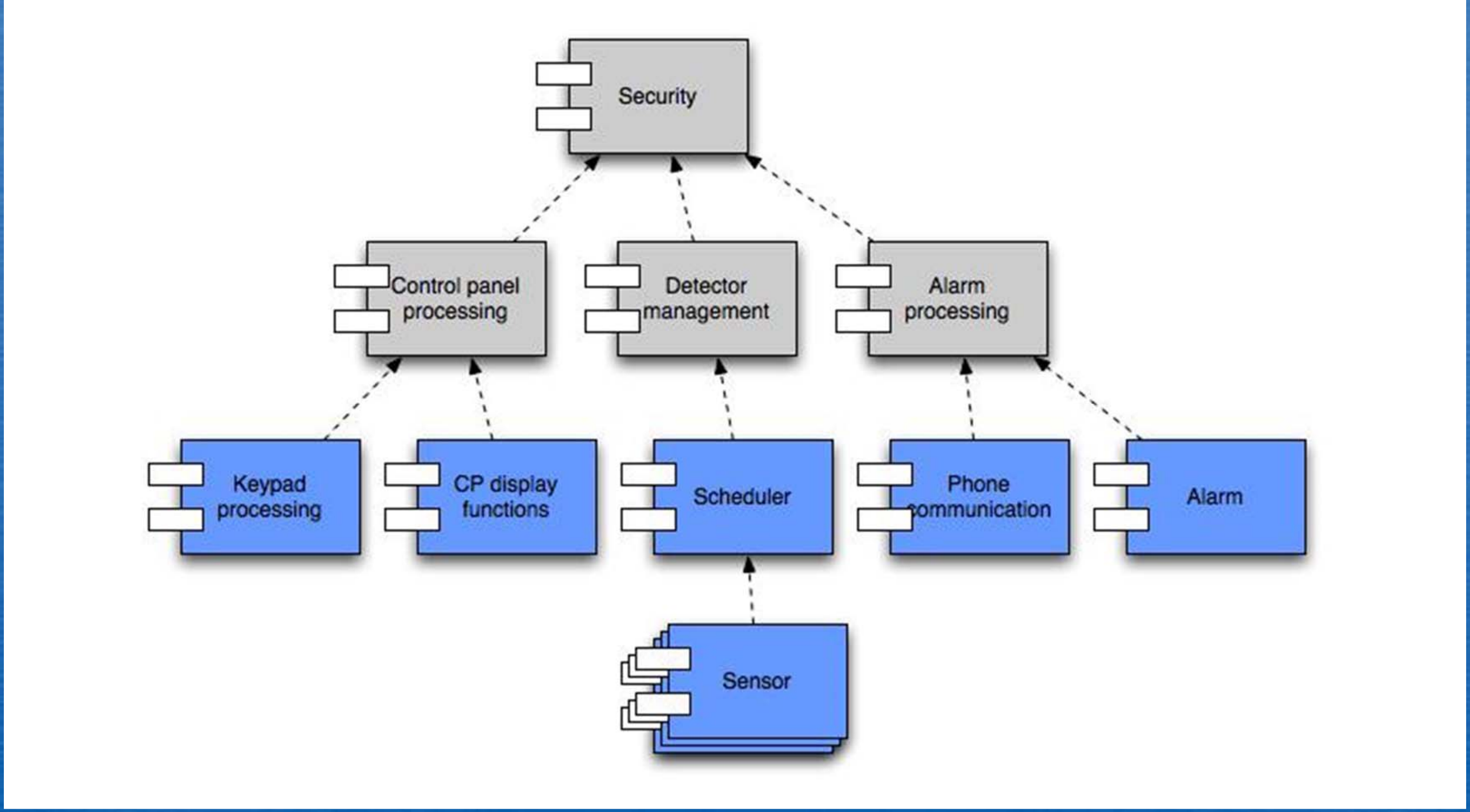
SafeHome Archetype



Component Structure



Component Elaboration


Security
Control panel processing
Detector management
Alarm processing
Keypad processing
CP display functions
Scheduler
Phone communication
Alarm
Sensor