Week 10 Software Architecture

- **Software Architecture**
  - Definition
    - The software architecture of a system is the set of structures needed to reason about the system, which comprises software elements, relations among them, and properties of both
- **Architectural Design**
  - Definition
    - Architectural design is the process for identifying the sub-system control and communication
    - The output of this design process is a description of the software architecture
  - Subsystem
    - Software architecture is about how a system being decomposed into subsystems (components) and how are they interface with each other
    - Definition
      - A subsystem is a grouping of elements that form part of a system
    - Characteristics
      - Provides major service
      - Highly cohesive
        - Contains highly related classes with respect to subsystem
      - Loosely coupled
        - Relatively independent of other subsystems
      - May be decomposed further into smaller subsystems
  - Abstraction
    - Architecture in the small
      - Individual programs -- decomposed into components
    - Architecture in the large
      - Complex enterprise systems that include other systems -- distributed over different computers
  - Advantages of Explicit Architecture
    - Stakeholder communication
      - Architecture may be used as a focus of discussion by system stakeholders
    - System analysis
      - Analysis of whether the system can meet its non-functional requirements is possible
    - Large-scale reuse
      - The architecture may be reusable across a range of systems
      - Product-line architectures may be developed
  - Use of architectural models
    - As a way of facilitating discussion about the system design
    - As a way of documenting an architecture that has been designed
- **Architectural View**
  - 4+1 View Model
    - A logical view
      - Key abstractions as objects or object classes
    - A process view
      - At run-time, the system is composed of interacting processes

- A developement view
  - How software is decomposed for developement
- A physical view
  - Shows system hardware, how software components are distributed across the processors in the system

Architectural Patterns

- Definition
  - An architectural pattern is a stylized description of good design practice, which has been tried and tested in different environments
  - A general, reusable solution to a commonly occurring problem in software archiecture with a given context
- Model-View-Controller (MVC) Pattern
  - Serves as a basis of interaction management in many web-based systems
  - Components
    - Model
      - Manages the data, logic and rules of the application
        - Core functionality and data
    - View
      - Can be any output representation of information, such as a chart or a diagram
        - Display the information to users
    - Controller
      - Accepts input and converts it to commands for the model or view, enables the interconnection between the views and the model
        - Handles the input from the user
  - Advantages
    - Allows the data to change independently of its representation and vice versa
  - Disadvantages
    - Involves additional code and code complexity
- Layered pattern
  - Usually usesd in building general desktop applications and relatively simple web apps
  - Advantages
    - Allows replacement of entire layers
  - Disadvantages
    - Performs poorly
    - Difficult to separate the layers
- Repository pattern
  - Advantages
    - Flexible architecture for data-intensive systems
  - Disadvantages
    - Difficult to modify repository since all other components are coupled to it

UML Diagrams in systems architecture (不重要)

- Package diagram
- Component diagram
- Deployment diagram