The Design Model, Architectural Views, and the SAD
The UP Design Model

- An object model of software components in the design.
  - Subsystems, layers, packages, classes, interfaces, events
- Before programming, an *inspiration* for the code base.
- If reverse-engineered from code, an abstraction or visualization of the code base.
- The *logical view* in the *Software Architecture Document* is a subset of the UP Design Model.
The UP Software Architecture Document (SAD)

- An important document in the UP.
  - “The Software Architecture Document provides a comprehensive architectural overview of the system, using a number of different architectural views to depict different aspects of the system.”

1. Architectural Representation
2. Architectural Goals and Constraints
3. Use-Case View
4. Logical View
5. Process View
6. Deployment View
7. Implementation View
8. Data View
9. Size and Performance
10. Quality
11. . . . .
The Design Model, SAD, and Iterative Development

- Remember! In iterative development, these artifacts are not create once, or fully in an early stage.
- Rather, they start small and incomplete.
- Especially through the Elaboration phase iterations, they grow and mature.
- They grow out of the *programming* experiences of each short iteration, not speculation.

Key idea:
- In iterative development, the architecture evolves through a combination of partial up-front designs and *programming* iterations.
The UP Architectural Views

- **Logical**
  - Subset of Design Model.
  - Architecturally significant design elements.
  - Subsystems, layers, packages, summary of frameworks.
  - Case study of a few key use case realizations.
    e.g., scenario interaction diagrams

- **Data**
  - If persistence is significant and not handled auto-
    magically, and requires considerable planning and work (e.g., schema mapping definition and maintenance).
The UP Architectural Views

- **Process**
  - All processes, and architecturally significant threads.
  - Interaction diagrams illustrating key scenarios of inter-process collaboration.

- **Deployment**
  - Physical network/hardware and the deployment of processes to nodes.
The UP Architectural Views

- **Implementation**
  - Subset of Implementation Model
  - Description of the mapping of important architectural elements (layers, subsystems, modules) into physical components
e.g., JAR files.

- **Use Case**
  - Subset of the Use Case Model
  - “Architecturally significant use cases”
  - Useful ???
In this seminar...

- We do not have time to fully explore the data, process, implementation, and deployment views.
- We will focus on the design model and its logical view.
- The UML includes notation to visualize all the views.
  - Class diagrams.
  - Notation for processes and threads.
  - Deployment diagrams.
  - Component diagrams.
  - . . .