

# CSC444 - Software Engineering I

Week 5-2

## Design

- Analysis vs. design (high- vs. low-level)
  - break it down to component
    - focuses on what ... req's analysis
  - describe how to implement the system
    - focuses more on how ... implementation
- Communication with stakeholders
- Project management

## Issues

- Constraints
  - hardware, software, standards
  - expertise of the team members
- Design methodology
  - set of heuristics and direction
  - notations for capturing design
  - mapping of design entities to implementation

## Design methodologies

- Procedural (function) abstractions
  - maps its input to output, may have side effects, may return a result
  - describes what a procedure (non-trivial) does not how it does it
  - need to specify constraints (e.g. requires, error checking)
- Data abstraction
  - identify data objects and operations performed on them
  - determine relationships among objects
    - class diagram

## Crash course on OO

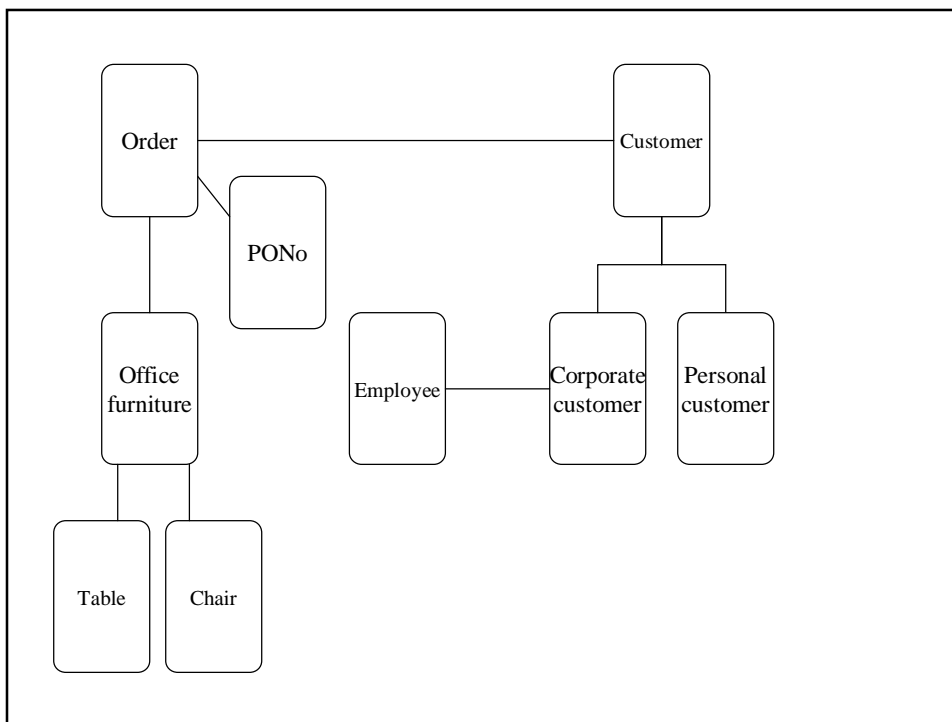
- An object is a data abstraction
  - represents part of a real world
  - is unique (or distinguishable) and has substance
  - object = identity + state + behavior
  - simple (atomic) vs. composite (Part\_of)
  - object communicate via message passing (method invocation)
- A class groups similar objects
  - classes can be form an inheritance hierarchy (ISA)
  - single vs. multiple inheritance

## OO methodologies

- Requirement analysis
  - use cases
  - indication on architectural style
- Analysis and Design
  - architecture design (system model)
  - identify objects, group in classes, determine class/objects relationships
- Implementation
  - map classes to entities of the language of choice

## What is needed for OO design

- 1. A class diagram depicting the static decomposition of the system
  - class definitions via its members
    - name, attributes, methods
  - ISA hierarchy (or inheritance)
  - class associations depicting relationships



## ... cont'd

- 2. A state diagram to model the dynamic behavior of a single object (an instance of a class)
  - FSM, StateCharts, state diagrams
  - StateCharts are extended FSM that have local variables
- 3. An interaction diagram to model the sequence of messages forming an interaction
  - sequence diagram: ordering of events
  - collaboration diagram: objects and their relationships

A state diagram describes the services provided by an object during its life cycle.

