Object oriented analysis and design

Module 3: Use-Case Modeling

Objectives

- Describe system behavior and show how to capture it in a model.
- Demonstrate how to read and interpret:
  - A use-case diagram
  - An activity diagram

Where Are We?

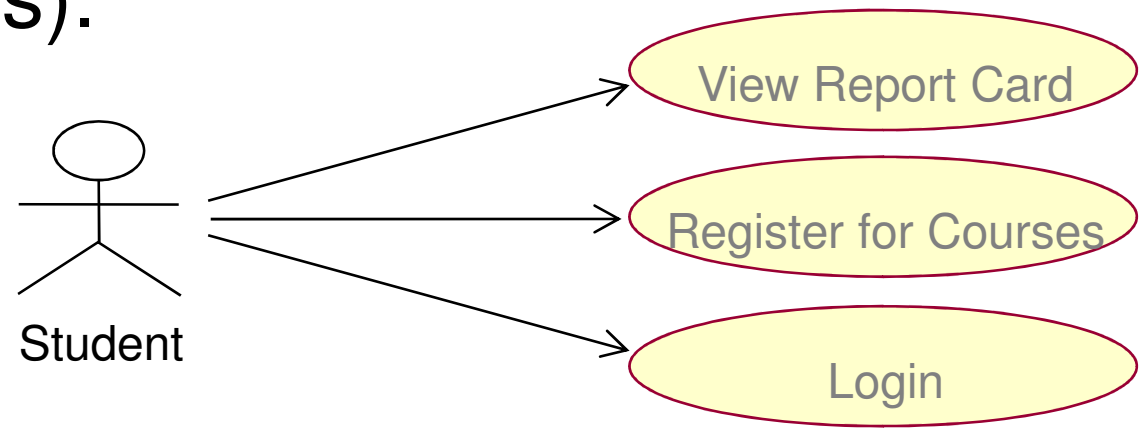
- Concepts in use-case modeling
- Use-case diagrams
- Activity diagrams

What Is System Behavior?

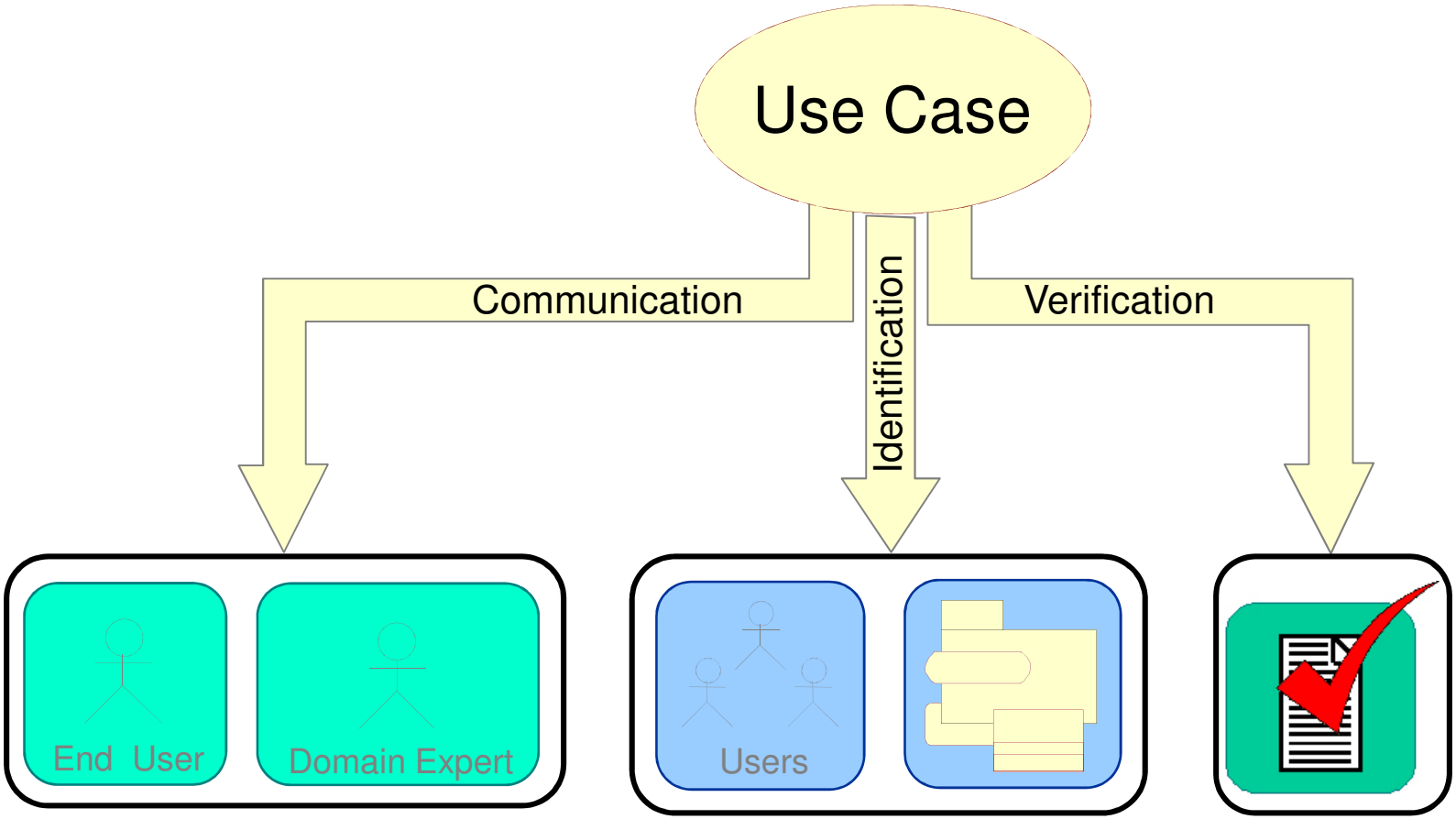
- System behavior is how a system acts and reacts.
  - It comprises the actions and activities of a system.
- System behavior is captured in use cases.
  - Use cases describe the interactions between the system and (parts of) its environment.

What Is a Use-Case Model?

- A model that describes a system's functional requirements in terms of use cases.
- A model of the system's intended functions (use cases) and its environment (actors).

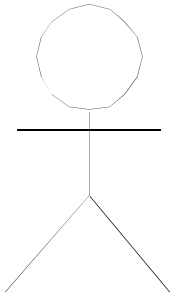


What Are the Benefits of a Use-Case Model?



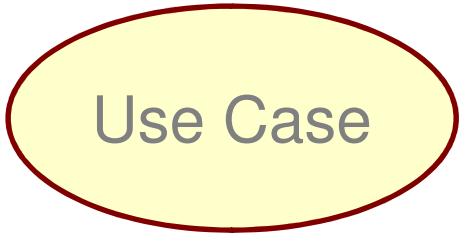
Major Concepts in Use-Case Modeling

- An actor represents anything that interacts with the system.



Actor

- A use case describes a sequence of events, performed by the system, that yields an observable result of value to a particular actor.

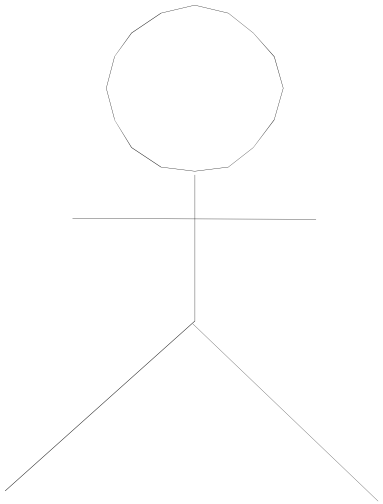


Where Are We?

- Concepts in use-case modeling
- Use-case diagrams
- Activity diagrams

What Is an Actor?

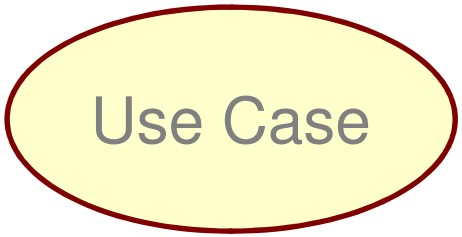
- Actors represent roles a user of the system can play.
- They can represent a human, a machine, or another system.
- They can actively interchange information with the system.
- They can be a giver of information.
- They can be a passive recipient of information.
- Actors are not part of the system.
  - Actors are EXTERNAL.



Actor

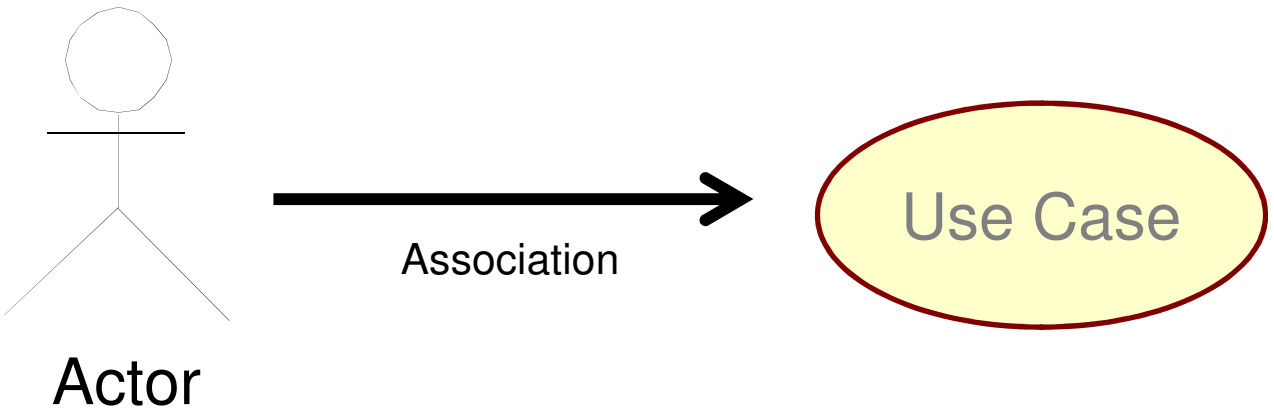
What Is a Use Case?

- Defines a set of use-case instances, where each instance is a sequence of actions a system performs that yields an observable result of value to a particular actor.
  - A use case models a dialogue between one or more actors and the system
  - A use case describes the actions the system takes to deliver something of value to the actor



Use Cases and Actors

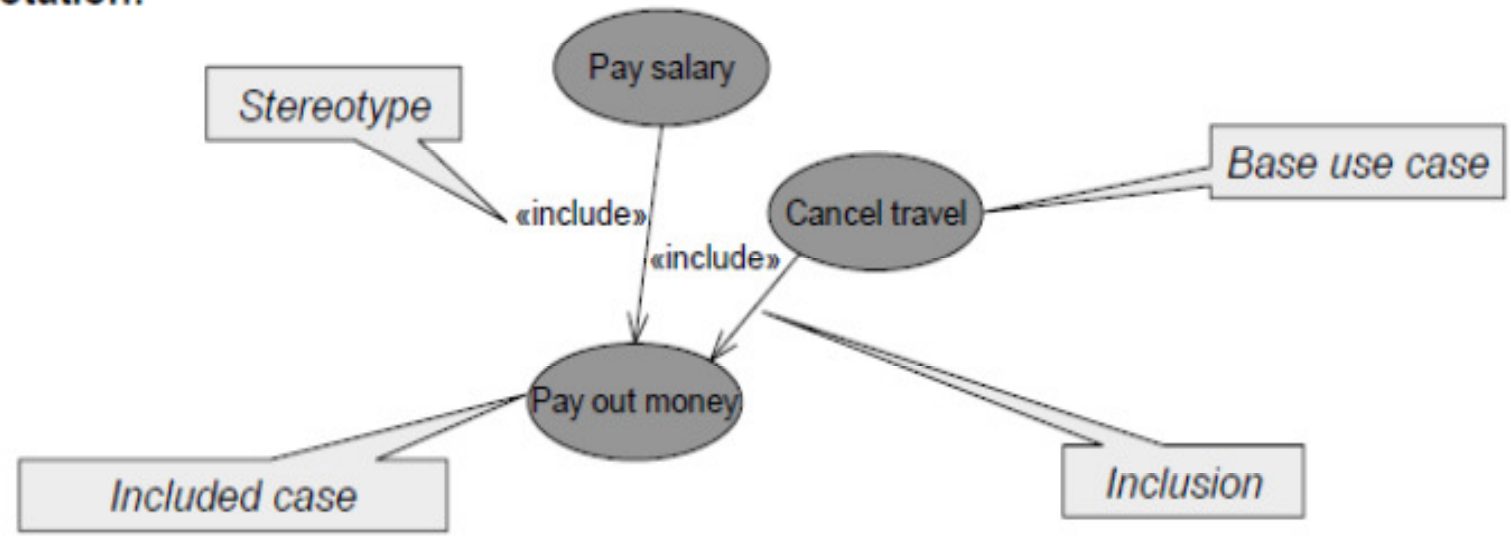
- A use case models a dialog between actors and the system.
- A use case is initiated by an actor to invoke a certain functionality in the system.



Include Use Cases

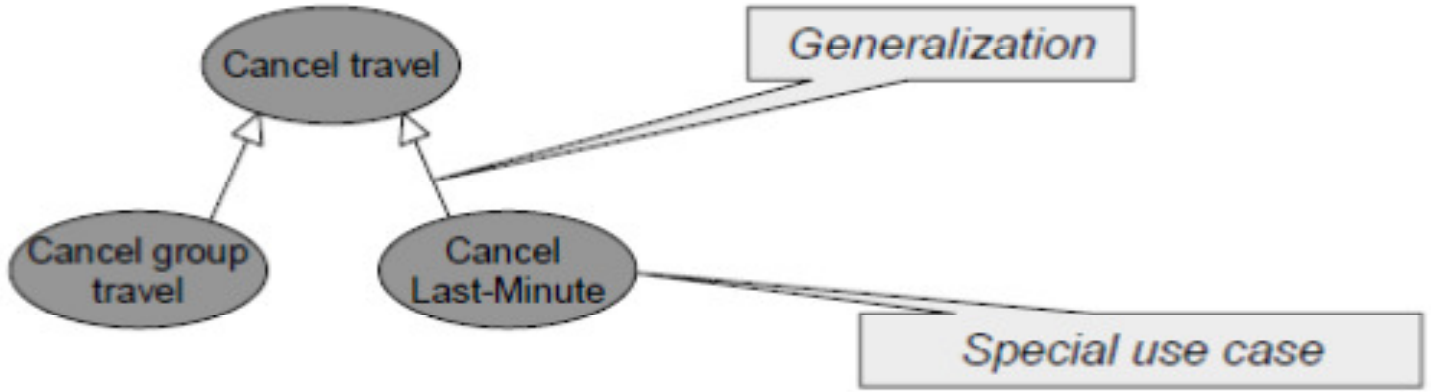
- **Include** means the inclusion of particular action sequences in the base use case.
- The included use case can be used independent of the base use cases.

Notation:



Generalization of Use Cases

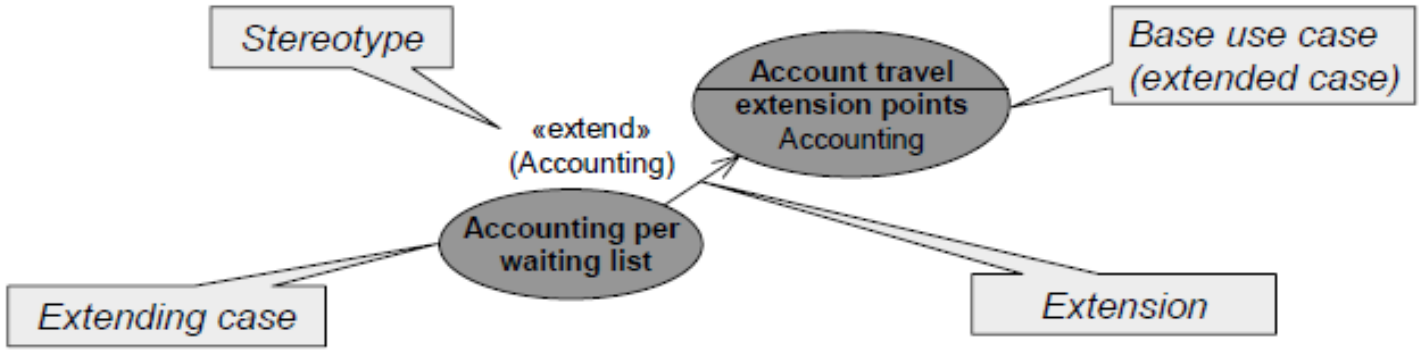
- **Generalization** correlates (taxonomically) more specialized use cases and more general ones. The special case inherits **every** property from the general and adds incrementally further properties or **replaces** them.



Extend Use Cases

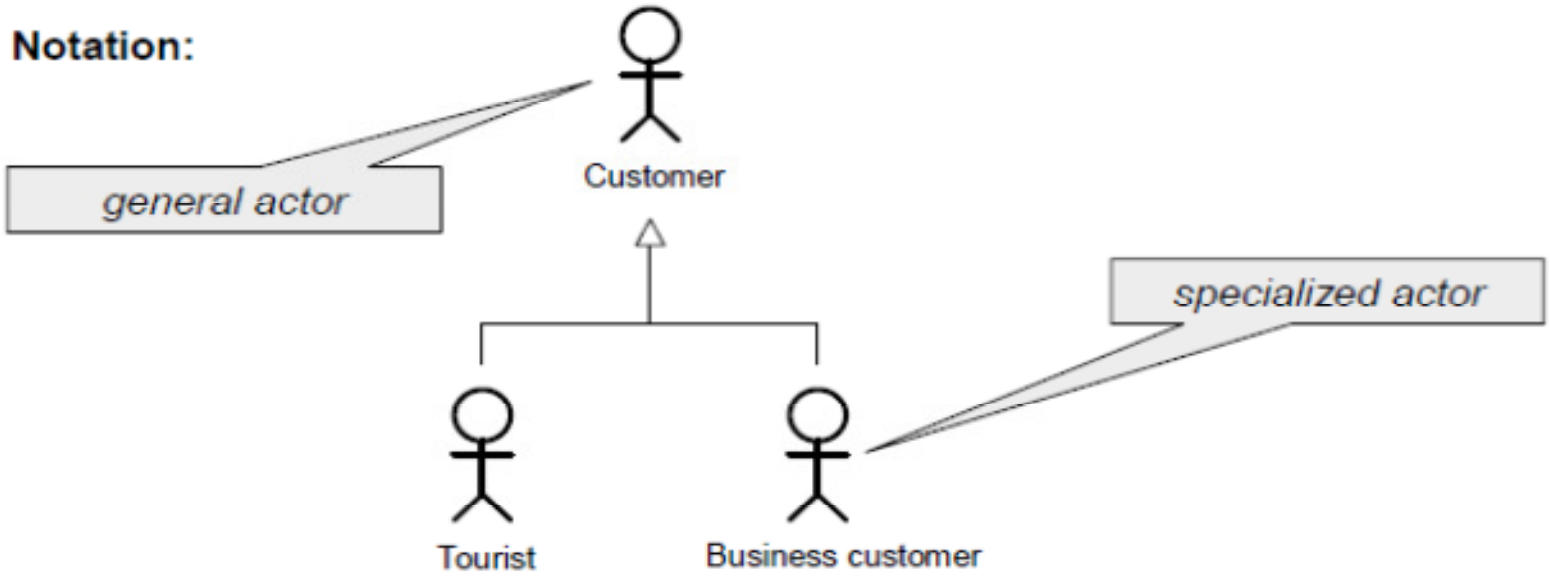
- **Extensions** define variations and special cases with the meaning, that the extending case completes the base use case, i.e., may be inserted in the behavior of the base case.
- Extensions are included, maybe optional, at **extension points**.
  - A use case may have many extension points
  - An extending use case may extend one or more of these extension points

Notation:

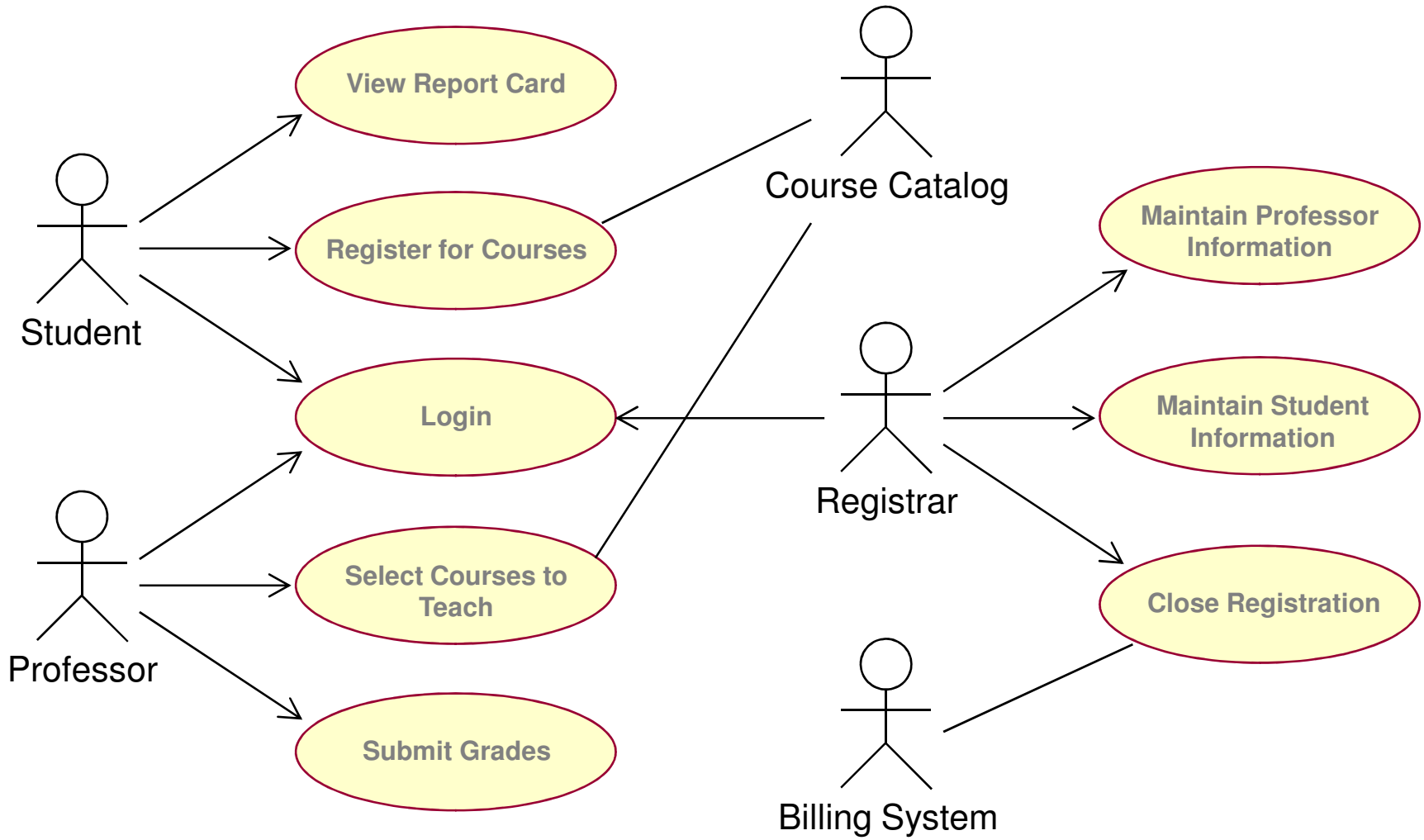


Generalization of Actors

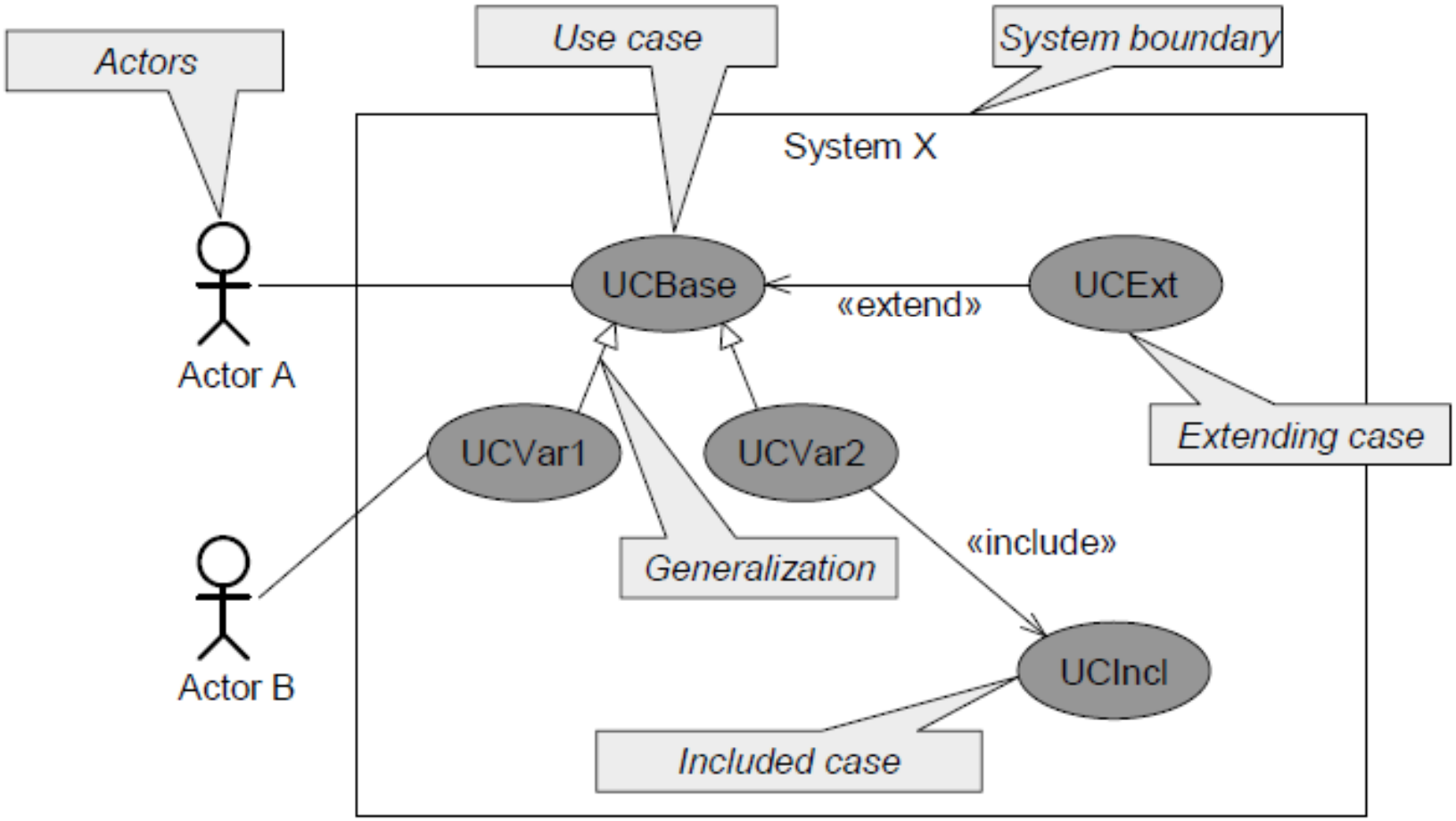
- Actors may have a **taxonomic relationship**.
- In use case diagrams the taxonomy is usually not shown. Separate actor diagrams show the relationships between actors.



How Would You Read This Diagram?



Summary of Concepts



Where Are We?

- Concepts in use-case modeling
- Use-case diagrams
- Activity diagrams

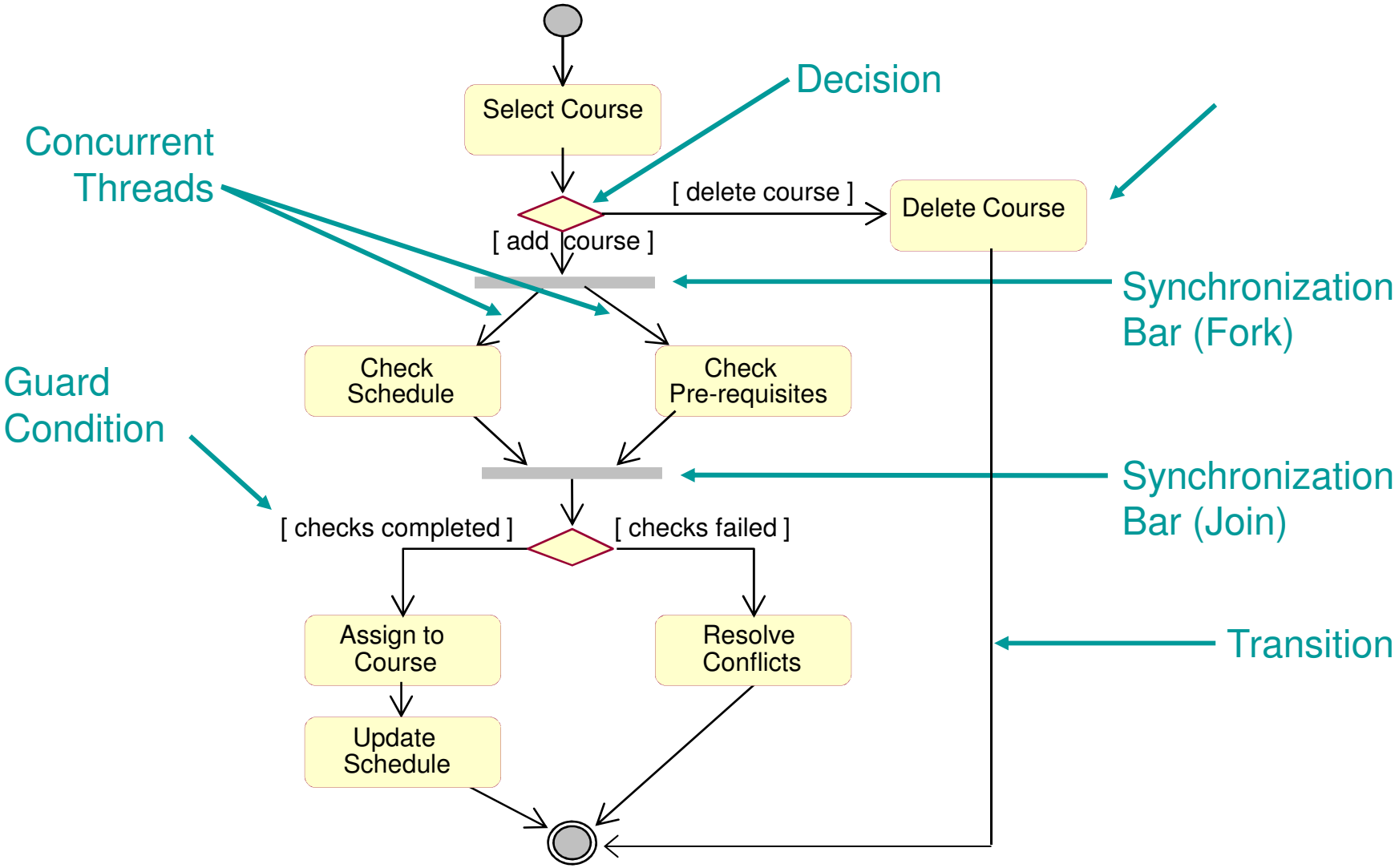
What Is an Activity Diagram?

- An activity diagram in the use-case model can be used to capture the activities and actions performed in a use case.
- It is essentially a flow chart, showing flow of control from one activity or action to another.

What Is an Activity?

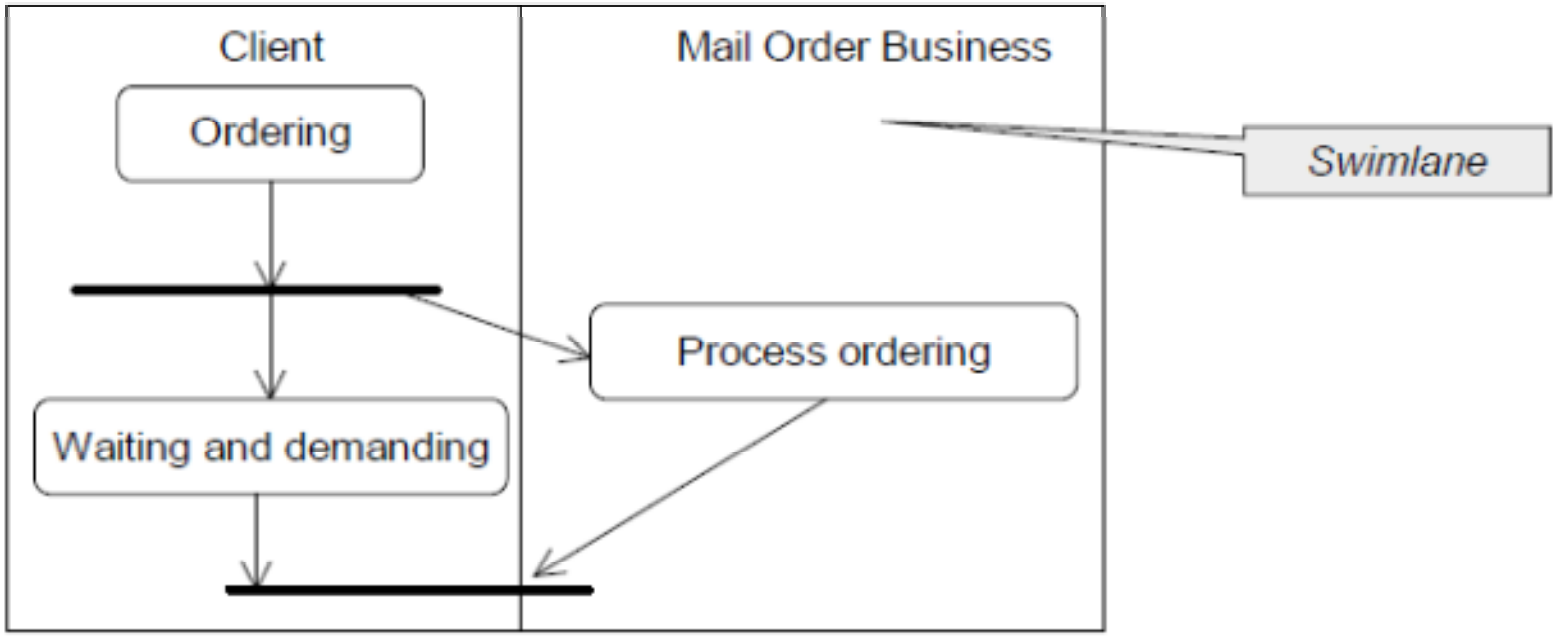
- A specification of behavior expressed as a flow of execution via sequencing of subordinate units.
  - Subordinate units include nested activities and ultimately individual actions.
- May contain boolean expression constraints when the activity is invoked or exited

Example: Activity Diagram



Swimlanes

- A **swimlane** shows the actions and activities being executed by a unit, an object or a class, mostly concurrent to other actions/activities.



Review

- What is system behavior?
- What is a use-case model? What are its benefits?
- What is an actor? A use case?
- What is an activity diagram?