Use Case Modeling

Chapter 7
Part of Requirements Modeling

Designing Concurrent, Distributed, and Real-Time Applications with UML
Hassan Gomaa (2001)

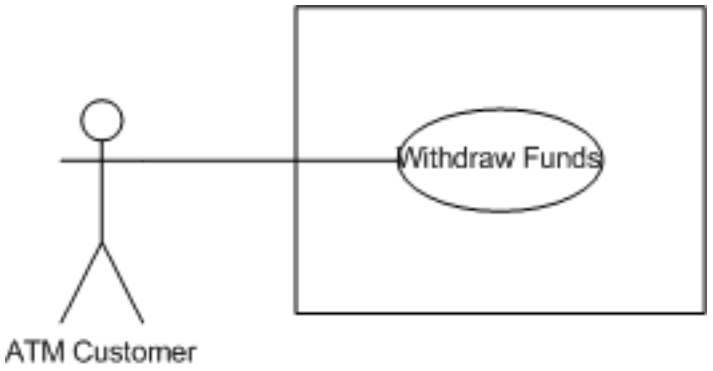
Use Case Model (7.1)

- A *use casediagram* consists of *actors* and *use cases*.
  - *Actors* are external users of the system.
  - *Use cases* are the things actors use the system for.
    - A sequence of interactions between one or more actors and the system.
- The *Use Case Model* shows the functional requirements of a system with *use case diagrams*.

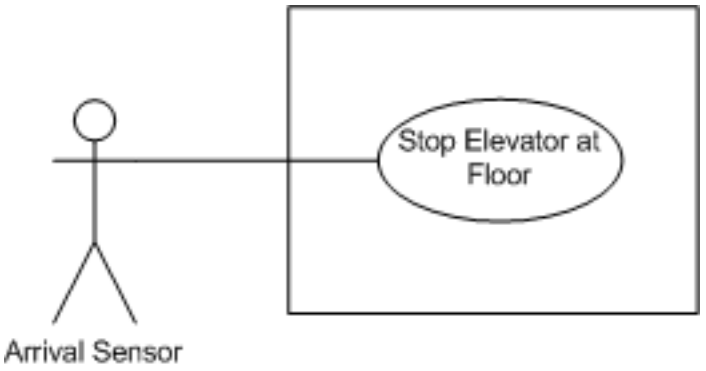
Actors (7.2)

- Represents a class of external user of the system.
  - Primary Actors initiate use cases.
  - Secondary Actors may also participate.
- Actors are not limited to humans
  - Any I/O device that connects to a system is an actor.
    - E.g. Sensors, timers, etc.
  - When the I/O is initiated or received by a human, the human is considered the actor.
    - E.g. keyboards, displays, etc.

Actor Examples (7.2)



Human Actor



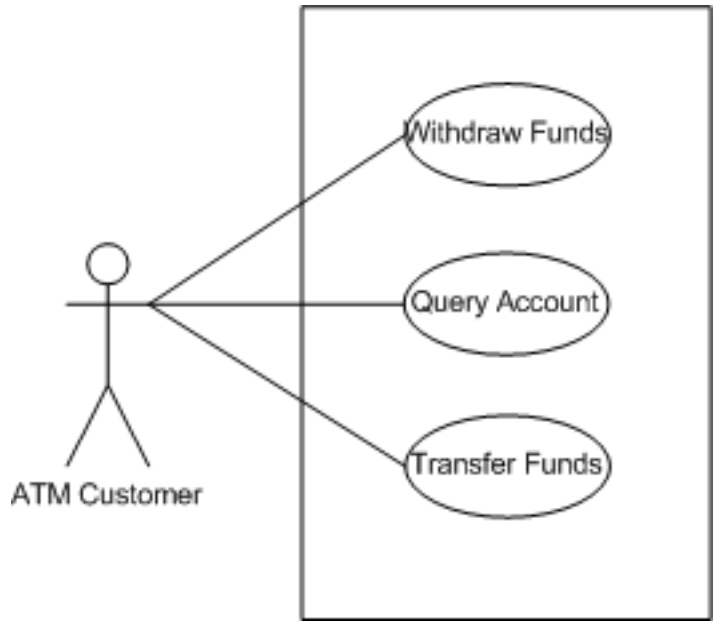
Non-Human Actor

Actors, Roles, and Users (7.3)

- Actors are used to represent a role played by a user.
  - Actors model types of users.
- Users are instances of a particular type
  - E.g. Driver, Navigator, Engineer, etc.
- A user can have multiple types.
- An actor can have multiple users.

Identifying Use Cases (7.4)

- Input from an actor are the starting places for use cases.
  - Start by considering the inputs to the system, and the sequences of events they will begin.
- Consider the ATM Customer, who may want to:
  - Withdraw funds
  - Query for account balances
  - Transfer funds from one account to another

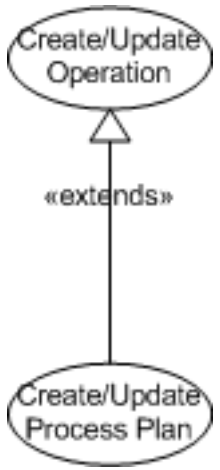


Use Case Documentation (7.5)

- In addition to the diagram, each use case should have a description consisting of:
  - Use Case Name
  - Summary
  - Dependency
  - Actors
  - Preconditions
  - Description
  - Alternatives
  - Post-condition
  - Outstanding Questions

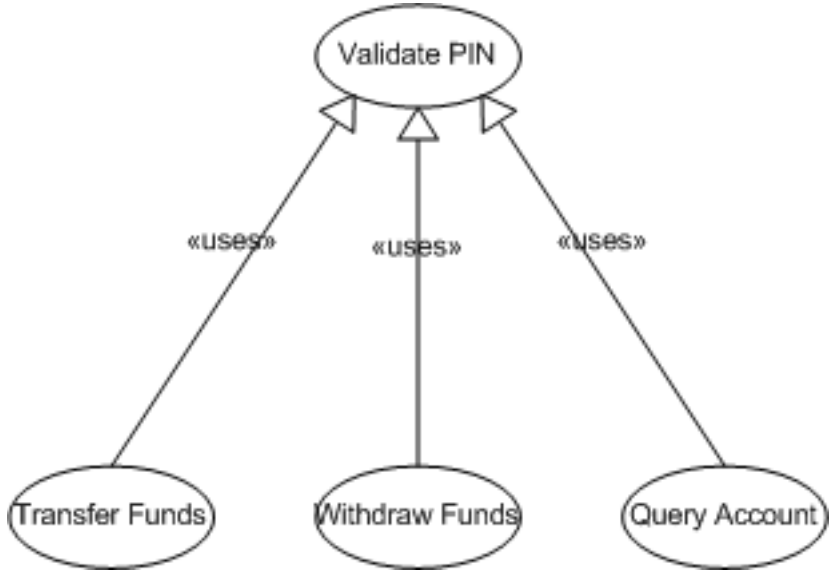
Use Case Relationships – Extend (7.7.1)

- Used to show alternative routes.
- The primary sequence of events is abstracted into a base use case, and extension use cases show the alternate routes.



Use Case Relationships – Include (7.7.2)

- Used to abstract common functionality among several use cases.
- Caution should be exercised: Abstract use cases should be used infrequently.
  - The goal is to model requirements, not functionally decompose the problem
- Called “uses” in VISIO



Use Case Packages (7.8)

- Like everything else in UML, use cases may be divided into packages.
- In COMET, use cases are grouped into packages by a common major actor, and the package is named after that actor.
  - E.g. FactoryOperatorUseCasePackage would contain use cases in which the factory operator was a major actor.

Summary (7.9)

- Use Cases are used to document functional requirements in COMET.
- Use Case Diagrams consist of Actors and Use Cases
- Each use case should have a use case description
- Use Cases can extend and include one another.