Chapter 5

Modeling System Requirements with Use Cases

User-Centered Development and Use-Case Modeling

User-centered development – a process of systems development based on understanding the needs of the stakeholders (mainly the users) and the reasons why the system should be developed.

Use-case modeling – the process of modeling system's functions in terms of business events, who initiated the events, and how the system responds to those events.

- Use-case modeling has roots in object-oriented modeling.
- Gaining popularity in non-object development environments because of its usefulness in communicating with users.
- Compliments traditional modeling tools.

System Concepts for Use-Case Modeling

Use case – a behaviorally related sequence of steps (scenario) for the purpose of completing a single business task.

- Description of system functions from the perspective of external users in terminology they understand.

Use-case diagram – a diagram that depicts the interactions between the system and external systems and users.

- graphically describes who will use the system and in what ways the user expects to interact with the system.

Use-case narrative – a textual description of the business event and how the user will interact with the system to accomplish the task.

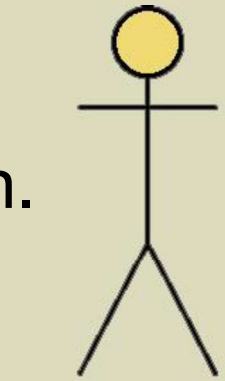
Basic Use-Case Symbols

Use case – subset of the overall system functionality

- Represented by a horizontal ellipse with name of use case above, below, or inside the ellipse.

Actor – anyone or anything that needs to interact with the system to exchange information.

- human, organization, another information system, external device, even time.


Actor Symbol

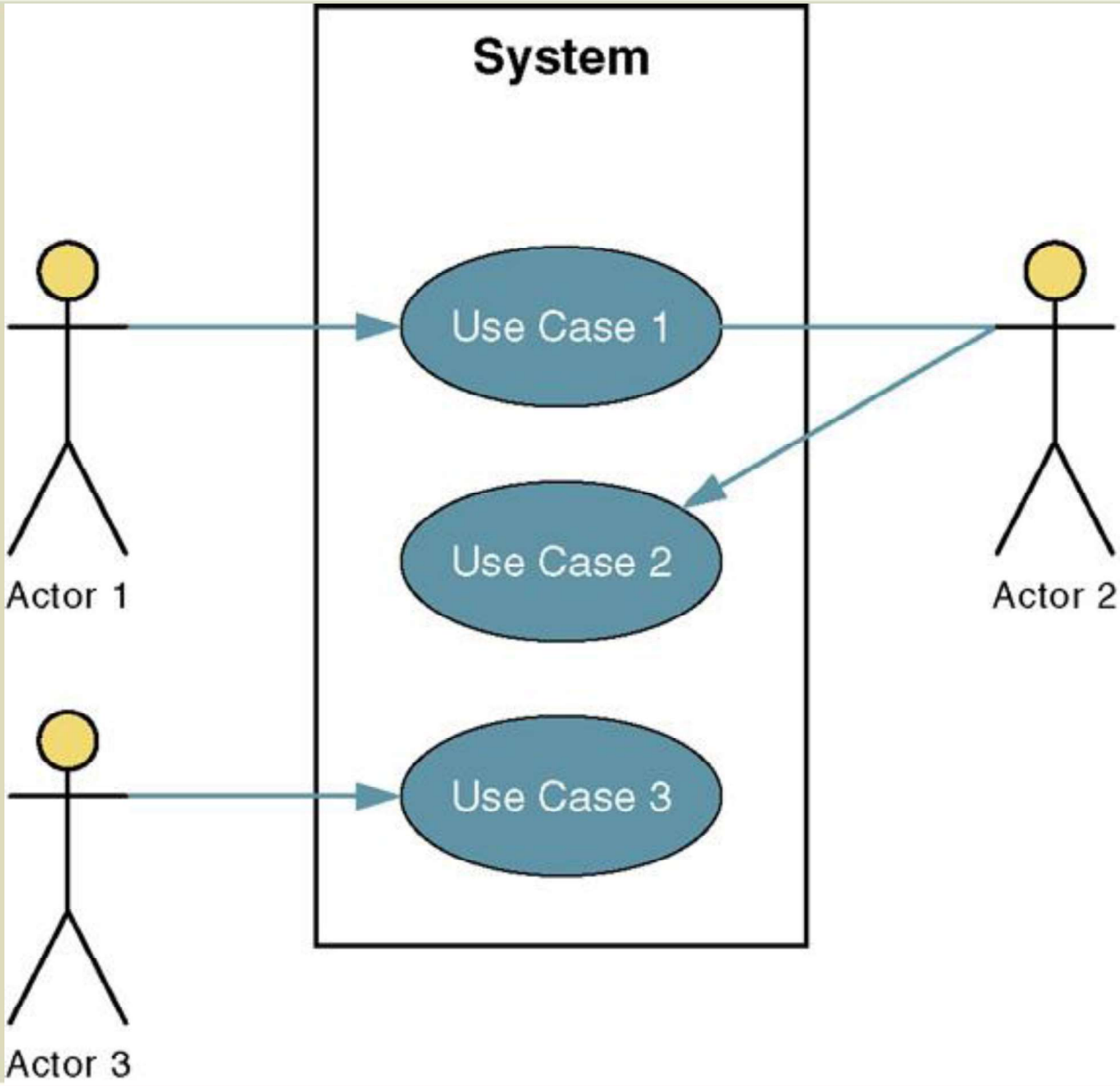
Temporal event – a system event triggered by time.

- The actor is time.

Four Types of Actors

- **Primary business actor**
  - The stakeholder that primarily benefits from the execution of the use case.
  - e.g. the employee receiving the paycheck
- **Primary system actor**
  - The stakeholder that directly interfaces with the system to initiate or trigger the business or system event.
  - e.g. the bank teller entering deposit information
- **External server actor**
  - The stakeholder that responds to a request from the use case.
  - e.g. the credit bureau authorizing a credit card charge
- **External receiver actor**
  - The stakeholder that is not the primary actor but receives something of value from the use case.
  - e.g. the warehouse receiving a packing slip

Sample Use-Case Model Diagram



Use Case Association Relationship

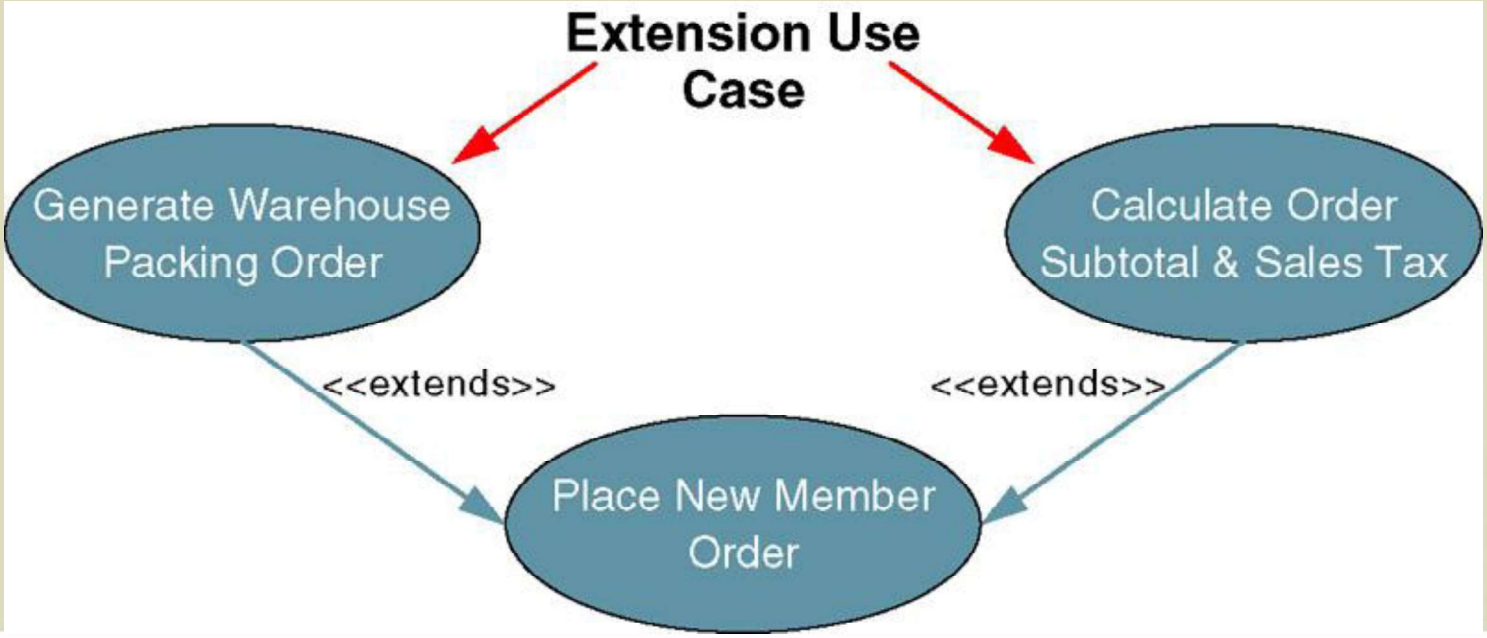
Association – a relationship between an actor and a use case in which an interaction occurs between them.

- Association modeled as a solid line connecting the actor and the use case.
- Association with an arrowhead touching the use case indicates that the use case was initiated by the actor. (1)
- Association lacking arrowhead indicates a receiver actor. (2)
- Associations may be bidirectional or unidirectional.

Use Case Extends Relationship

Extension use case –use case consisting of steps extracted from another use case to simplify the original.

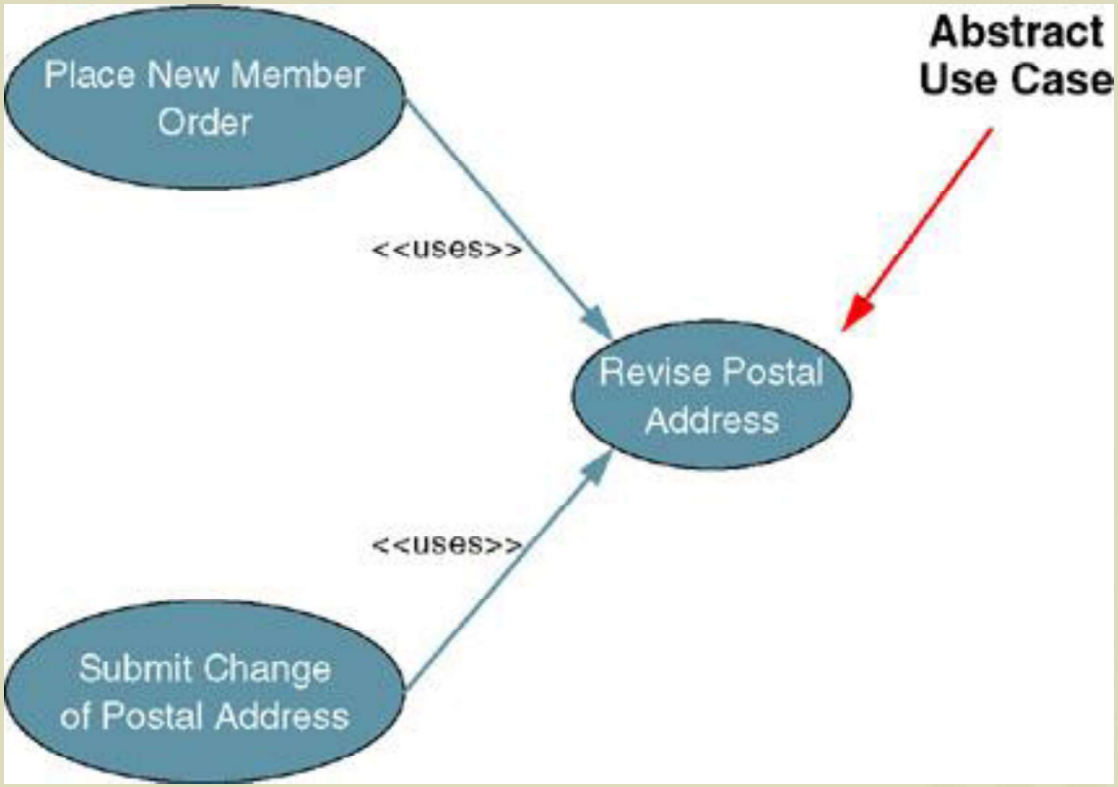
- Extends the functionality of the original use case.
- Generally not identified in the requirements phase
- Extends relationship represented as arrow beginning at the extension use case and pointing to use case it is extending.
- Labeled <<extends>>.



Use Case Uses Relationship

Abstract use case – use case that reduces redundancy in two or more other use cases by combining common steps found in both.

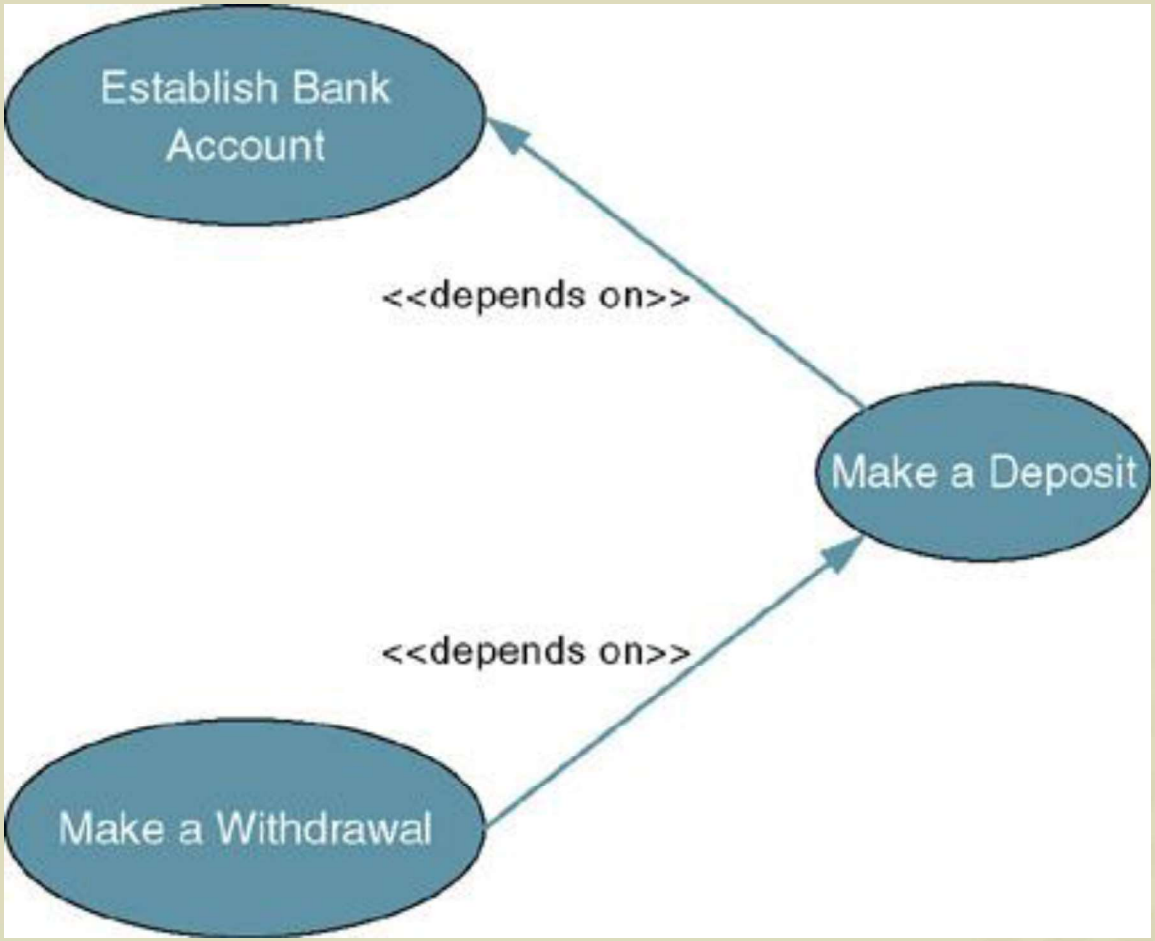
- Available by any other use case that requires its functionality.
- Generally not identified in requirements phase
- Relationship between abstract use case and use case that uses it is called a *uses* (or *includes*) relationship.
- Depicted as arrow beginning at original use case and pointing to use case it is using.
- Labeled <<uses>>.



Use Case Depends On Relationship

Depends On – use case relationship that specifies which other use cases must be performed before the current use case.

- Can help determine sequence in which use cases need to be developed.
- Depicted as arrow beginning at one use case and pointing to use case it depends on.
- Labeled <<depends on>>.

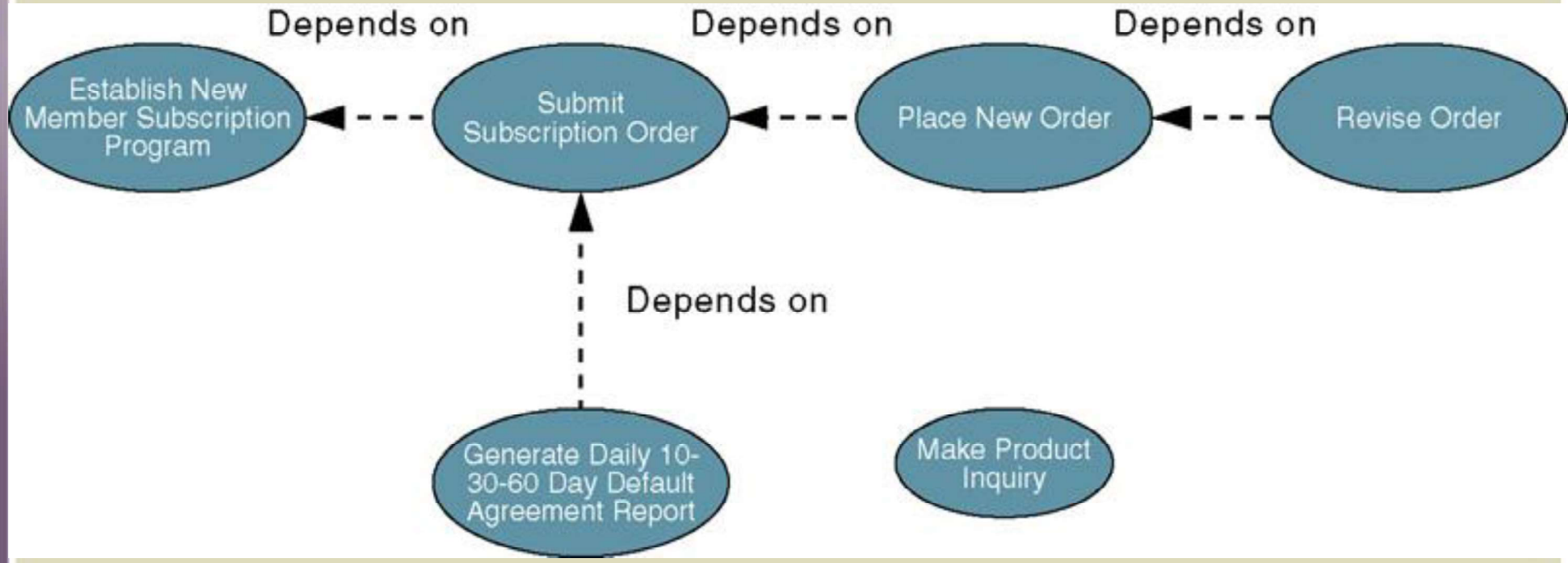


Use-Cases and Project Management

Use-Case Dependency Diagram – graphical depiction of the dependencies among use cases.

- Provides the following benefits:
  - Graphical depiction of the system's events and their states enhances understanding of system functionality.
  - Helps identify missing use cases.
  - Helps facilitate project management by depicting which use cases are more critical.

Sample Use-Case Dependency Diagram

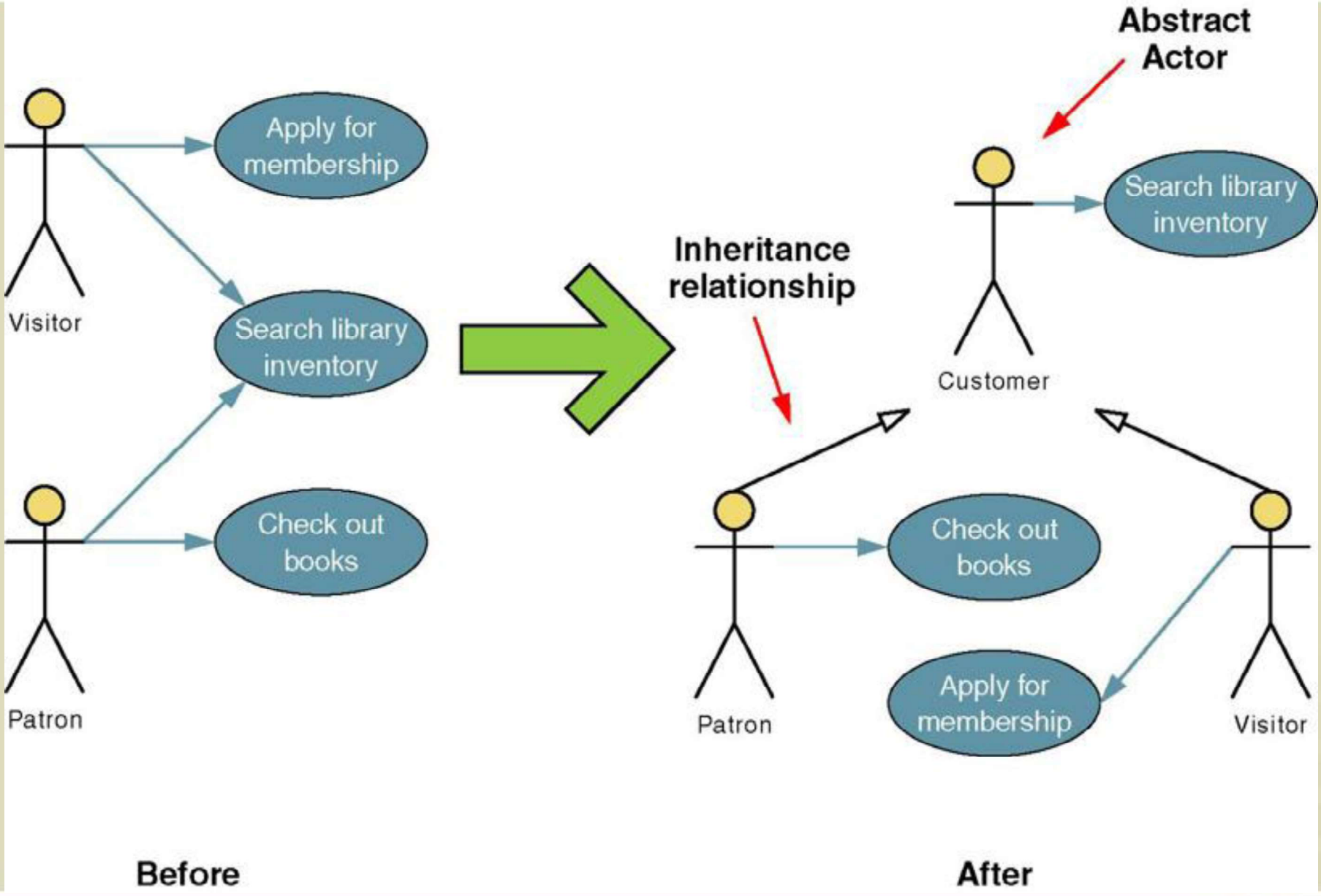


Use Case Inheritance Relationship

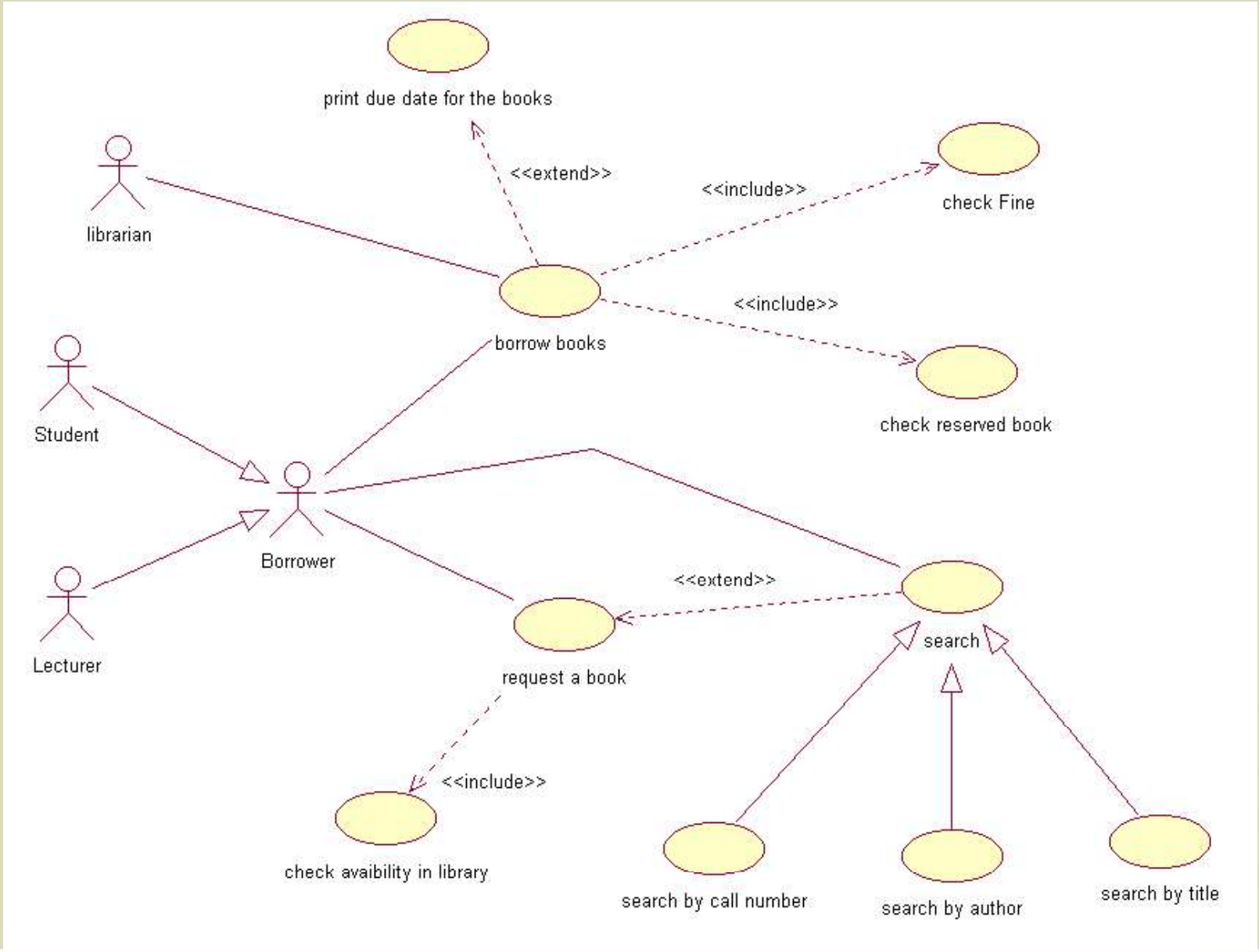
Inheritance – a use case relationship in which the common behavior of two actors initiating the same use case is extrapolated and assigned to a new *abstract* actor to reduce redundancy.

- Other actors can inherit the interactions of the abstract actor.
- Depicted as an arrow beginning at one actor and pointing to the abstract actor whose interactions the first actor inherits.

Use Case Inheritance Relationship



Use Case Diagram Example 1



Use Case Diagram Example 2