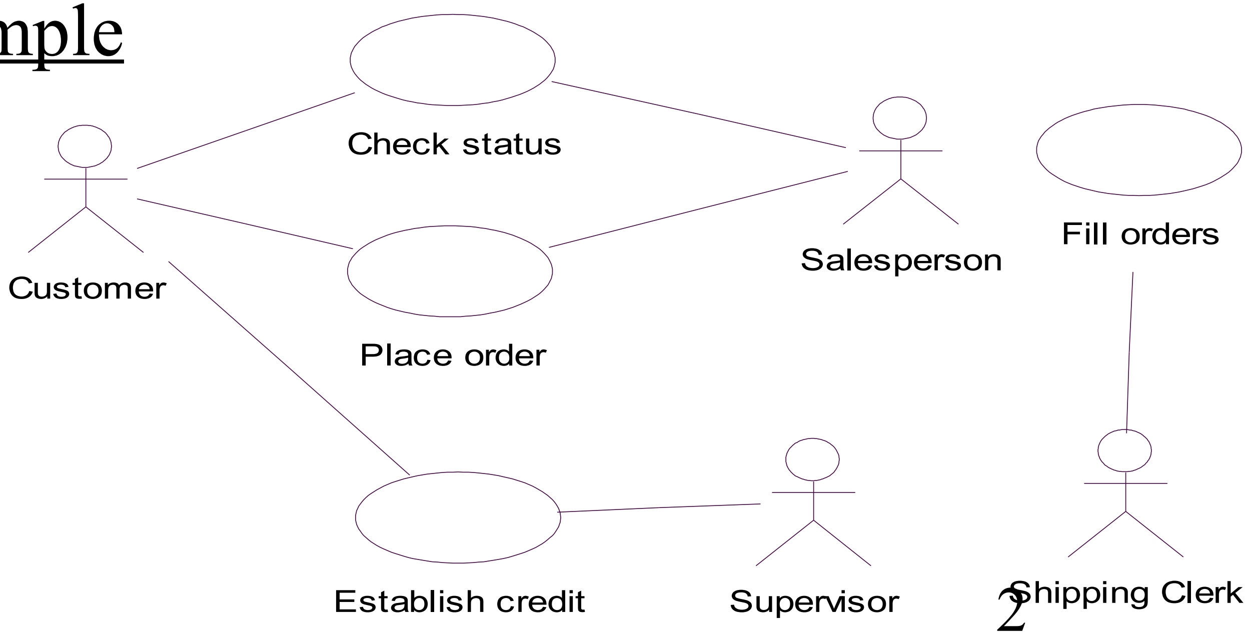
Use Case Diagrams

- **Use Case Diagrams**
- **Use Case**
- **Actor**
- **Use case description**
- **Use case realization (Scenario)**
- **Use case relationships**
  - **Extends**
  - **Uses**

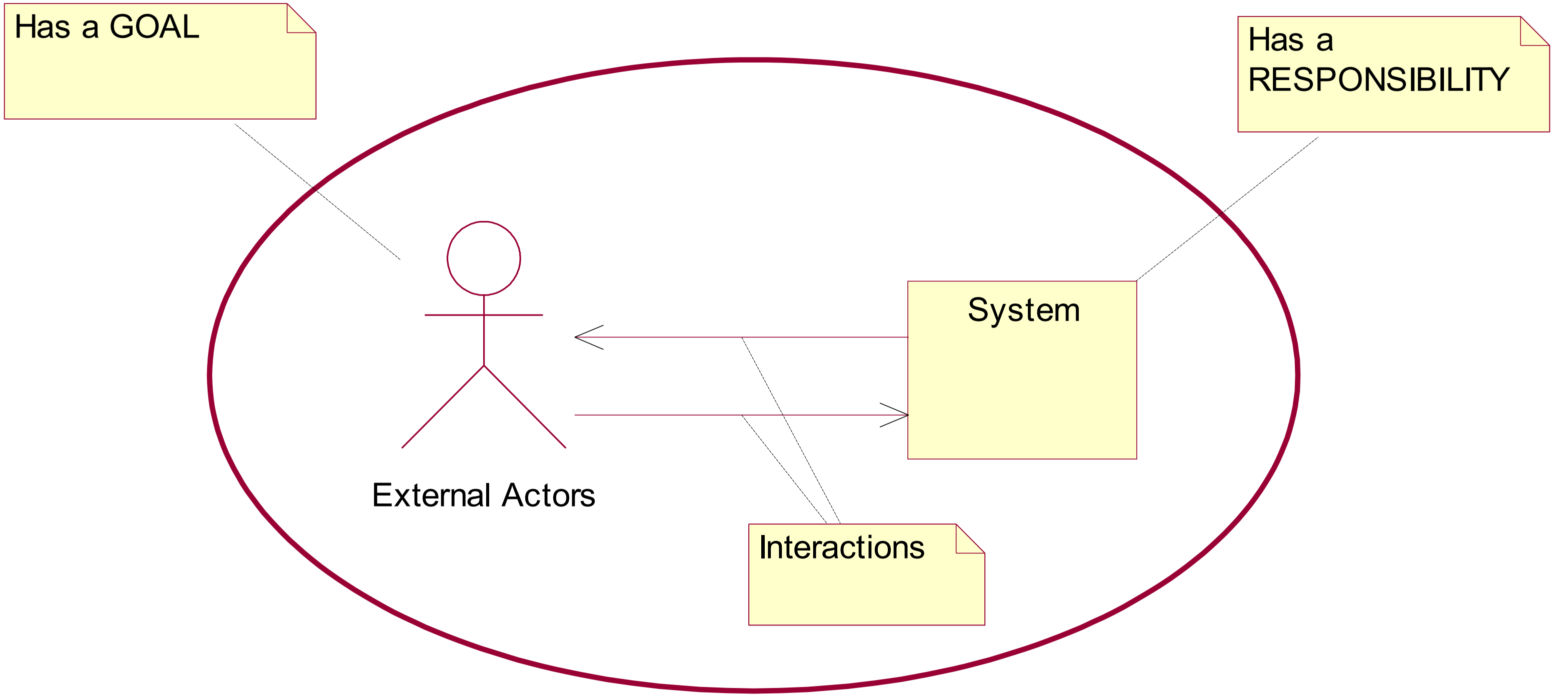
Use Case Diagrams

- shows the relationship among actors and use cases within a system.
- the use case model represents functionality of a system or a class as manifested to external interactors with the system
- shows a system boundary

Example



Use Case Model



Use Case

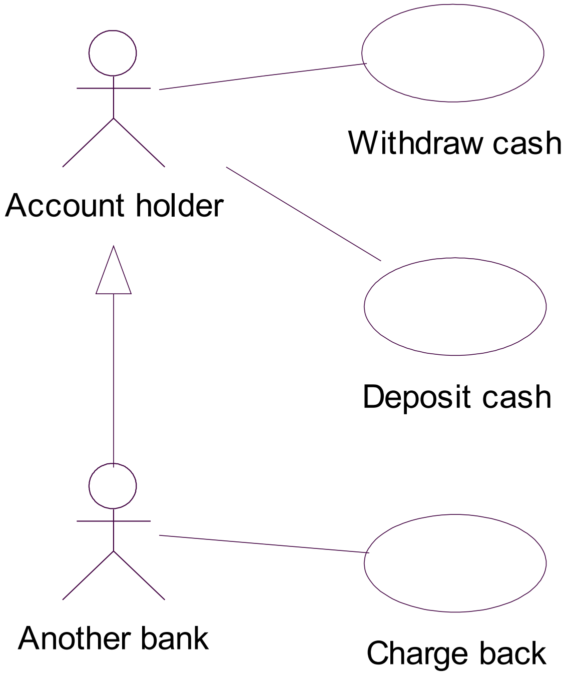
Use Case

- a sequence of transactions in a system whose task is to yield a measurable value to an individual actor of the system
- UML definition : a coherent unit of functionality provided by a system as sequences of messages among the system and one or more outside interactors (called *actors*) together with actions performed by the system
- the set of all use cases describes the whole functionality of the system
- represents WHAT the system must provide, NOT HOW

Actor

- a ROLE of object or objects OUTSIDE of a system that interacts directly with it as part of a use case
- one physical object may play several roles (modeled by several actors)
- types of actors
  - a primary actor initiates a use case
  - secondary actors (usually machines) are called by the system to complete a use case
- actor inheritance
  - the specialized actor gets all capabilities of the parent actor

Example (of actors and use cases)



- **Goal**
  - should be a central notion during the use case analysis
  - most people (even the non-technical ones) usually think in goals while making their jobs
  - main objective of a software systems is to provide “some assistance” on the users’ jobs
  - speak with their own language, the language of
    - GOALS and RESPONSIBILITIES.

- **Use case description**
    - Name
    - Goal
    - Preconditions
    - Postconditions (success and failed)
    - Normal flow of events (main scenario)
    - Alternative flows of events (alternative scenarios)
    - Additional properties
        - priority, schedule, time, frequency
        - other non-functional requirements
    - Open issues

Use case realization (Scenario)

- an instance of a use case (with real data values)
  - potentially hundreds to thousands in an application
- use cases represents a set of potential scenarios
- interaction diagrams describe how use cases are realized as interactions among societies of objects

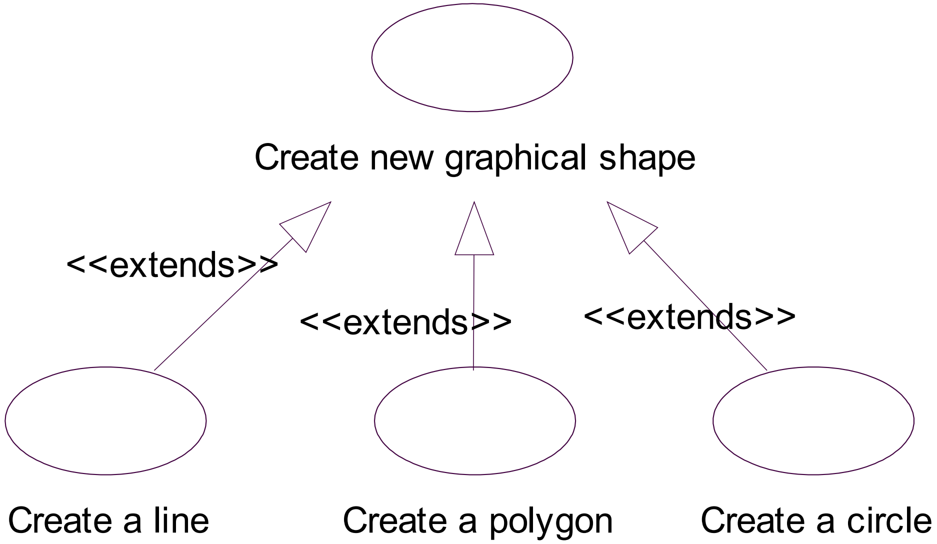
Use case relationships

- Communicates
  - the participation of an actor in a use case
  - the only relationship between actors and use cases
- Extends & Uses
  - are means for structuring and simplifying functional requirements

Use case relationships - Extends

- indicates that the instance of extended (base) use case MAY include the behavior specified by the extending use case
- a single use case may have several extenders
- extension often represents unusual behavior (exception, error)
- an *extension point* is a location within a use case at which action sequences from other use cases may be inserted.
- the base use case may be unaware of the extension

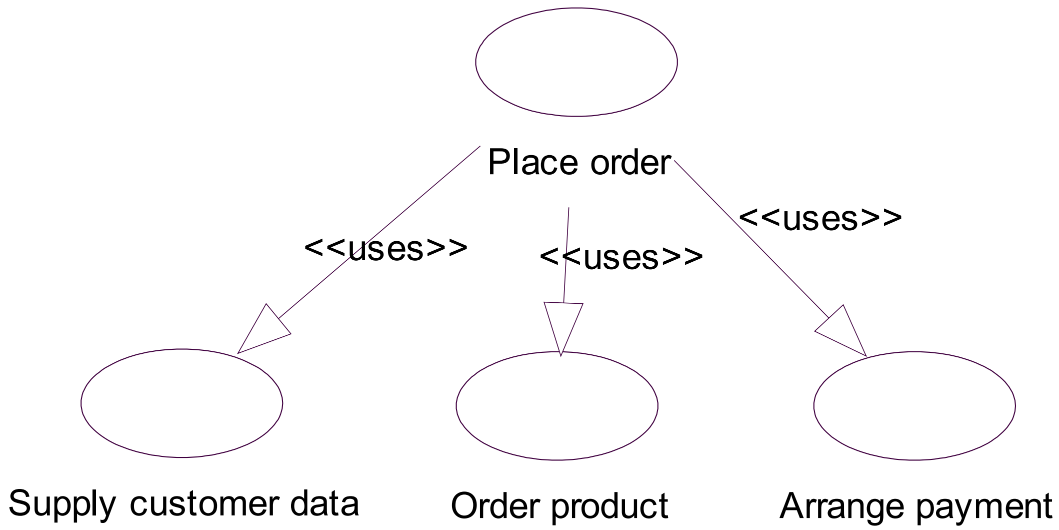
Example (of extends relationship)



Use case relationships - Uses

- indicates that an instance of the use case will also include the behavior as specified by the included use case(s)

Example



- **Process of use case modeling**
  - capture common vocabulary (glossary)
  - find actors (differentiate primary and secondary actors)
  - for each actor determine a set of its of use cases
  - briefly describe use cases and actors
  - package use cases and actors
  - structure the use case model
  - prioritize use cases
  - describe use cases