Requirements

The UML

Use-Case-Driven Approach

UML vs. Requirements Modeling

- UML: Unified Modeling Language
- A software analysis and design methodology mainly based on diagrams
- Requirements Modeling in UML:
  The Use-Case-Driven Approach
- Use cases are used to describe the outwardly visible requirements of a system
- They can be used later on in system design
- Developed by Booch, Jacobson and Rumbaugh of Rational Software (www.rational.com)

The Process

- Inception Phase:
  - Project description agreement
  - Project risks
  - Context of the project
  - Scope of the project
- Elaboration Phase:
  - Detailed definition of all use cases
  - UML diagrams modeling scenarios
  - Use case diagram(s)

Inception Phase

Project Description

- Project description agreement
  - Identify the problem and its root causes
  - Write a short textual description of the problem to be solved, and the key features of the system
  - Should not describe solutions
  - From a paragraph to a couple of pages for a complex project
  - Every stakeholder has to agree on the project description
- Project risks
  - Look at the system from many viewpoints
    - Other systems, marketing, technology, users, managers
  - Identify things that can go wrong
    - User resistance, inexperienced developers, system dependencies

Context of the Project

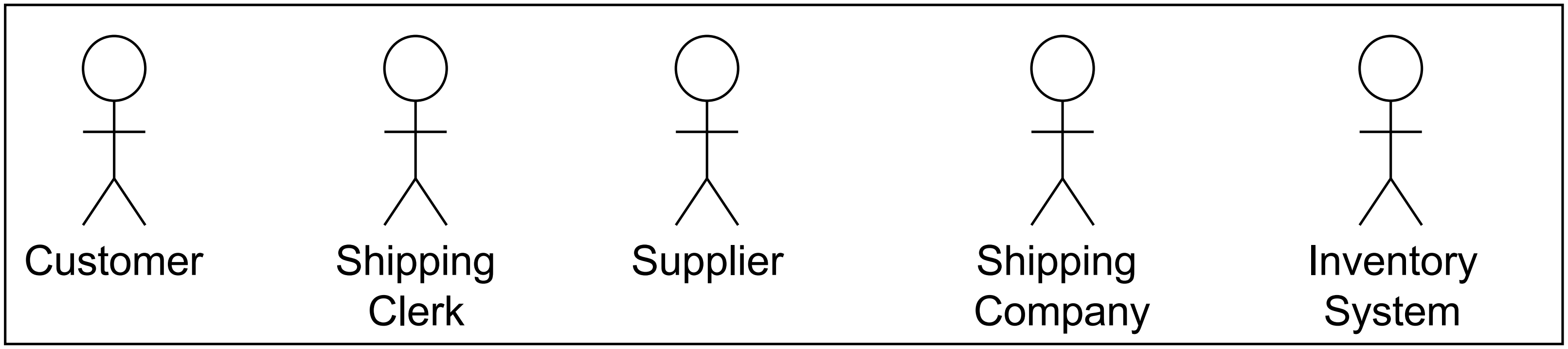
- Define what is inside the system, or system functionalities
  - Represented as use cases in the UML
- Define what is outside the system and interacts with the system
  - Represented as actors in the UML

Context of the Project

- Identify actors on the system
  - An actor is represented by its role, not its individuality
  - Actors are always external to the system
    - Users
    - Other software systems
    - Hardware devices
    - Data stores

Context of the Project

- Describe actors.
  - Customer: a person who orders products through the system.
  - Shipping company: UPS, FedEx, Purolator.
  - Shipping clerk: user of the system who packages, labels and ships orders.
  - Inventory system: software that tracks the company inventory.

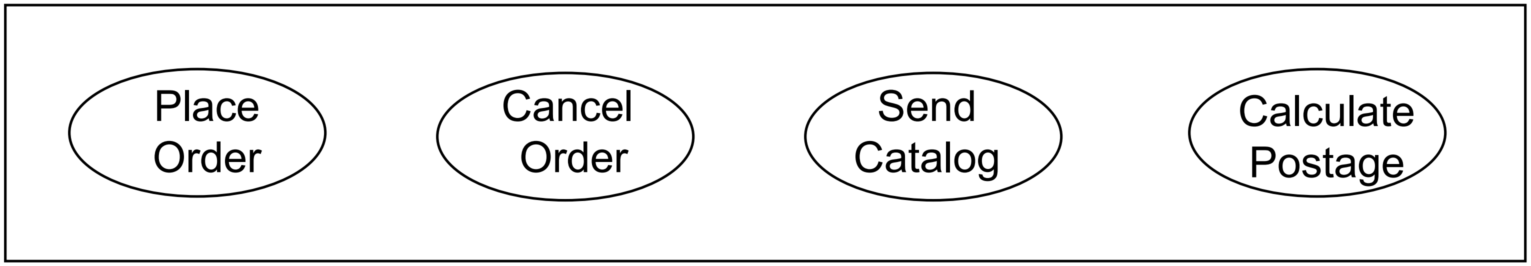


Context of the Project

- Identify use cases
  - What are the services used by the actors?
  - Who stores, accesses or deletes information in the database?
  - Startup, shutdown, diagnostics, installation
  - Maintenance
- Go through all the actors and identify how they can use the system

Context of the Project

- Order-processing use cases
  - Customer: place order, send catalog, get status on order, return product, cancel order, register complaint
  - Shipping clerk: print mailing labels, calculate postage
  - Inventory system: give product information, update product quantities



Scope of the Project

- Estimate what could realistically be implemented considering factors such as:
  - Time frame available
  - Budgetary envelope
  - Physical resources available
- The system description, risk analysis and assumptions must be met
- End of the inception phase
- Next step: adding details and structure

Elaboration Phase

Define Use Cases

- Use case: A coherent unit of externally visible functionality provided by a system unit.
- Used to define a behavior without revealing the internal details.
- A use case describes what the system does, not how it does it.
- Scenario: flow of events describing how a use case is realized.
- Each use case has a primary scenario.
- Eventually also has a set of alternate scenarios.
- Pre-conditions and post-conditions are stated.

Define Use Cases

Place Order

Pre-conditions:

A valid user has logged into the system

Primary Flow of Events:

1. (start) The customer selects Place Order
2. The customer enters its address
3. The customer enters the product codes it wants to order
4. The system provides the items description and prices, and a running total
5. The customer enters its credit card number
6. The customer clicks on submit
7. The system validates the information, saves the order and forwards the transaction request to the accounting system
8. (end) When the payment is confirmed, the order is marked as paid

Alternate Flow of Events 1:

In step 7, the system prompts the user to correct any incorrect information

Alternate Flow of Events 2:

In step 8, if the transaction is refused by the bank, the order is marked as pending

Post-conditions:

The order has been saved in the database

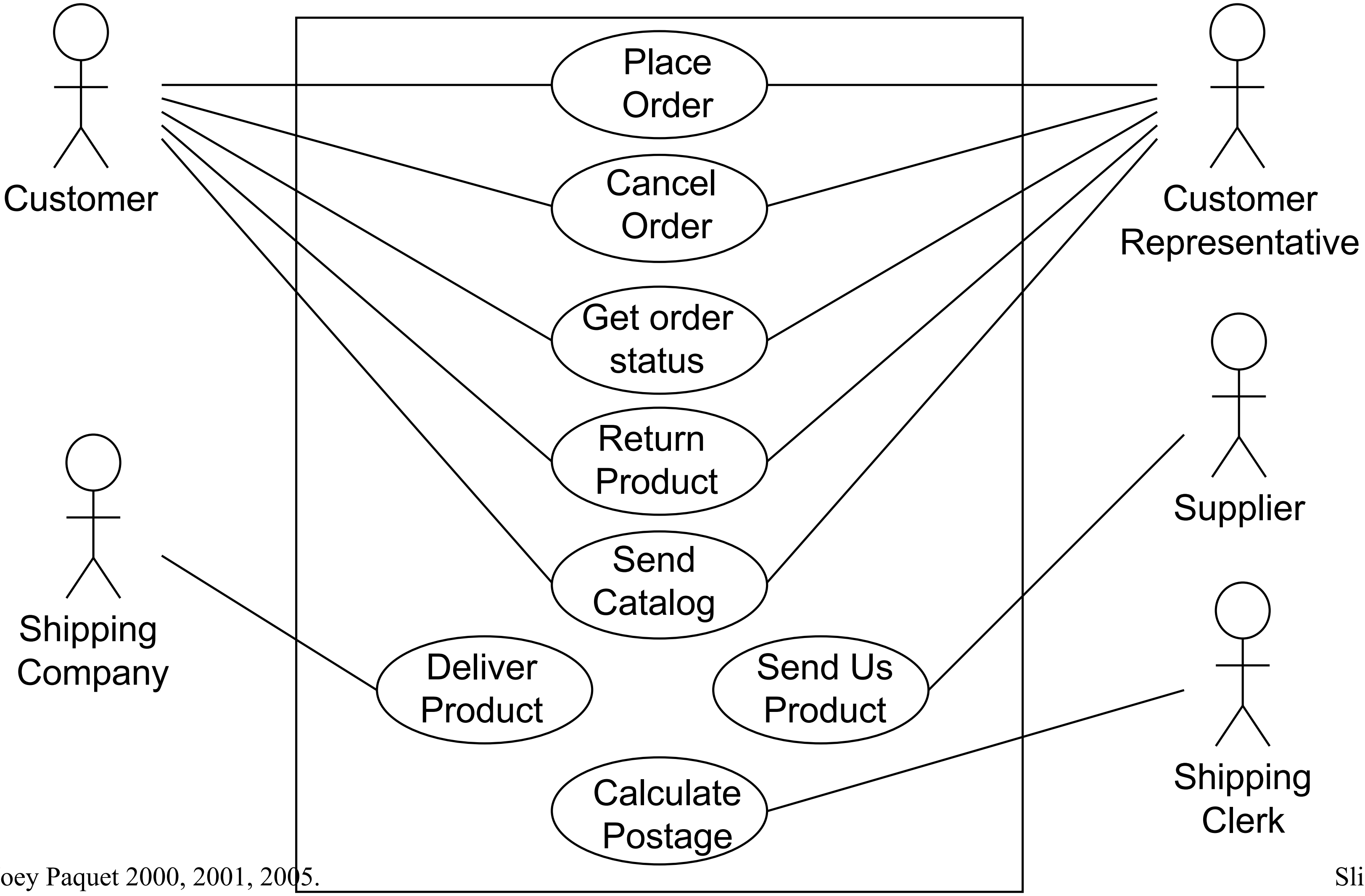
Scenarios: Diagrams

- Complex scenarios are better expressed using diagrams.
- The UML provides two kinds of diagrams:
  - Activity diagrams for a high-level description.
  - Sequence diagrams for more in-depth analysis.
- Will be covered in the tutorials.

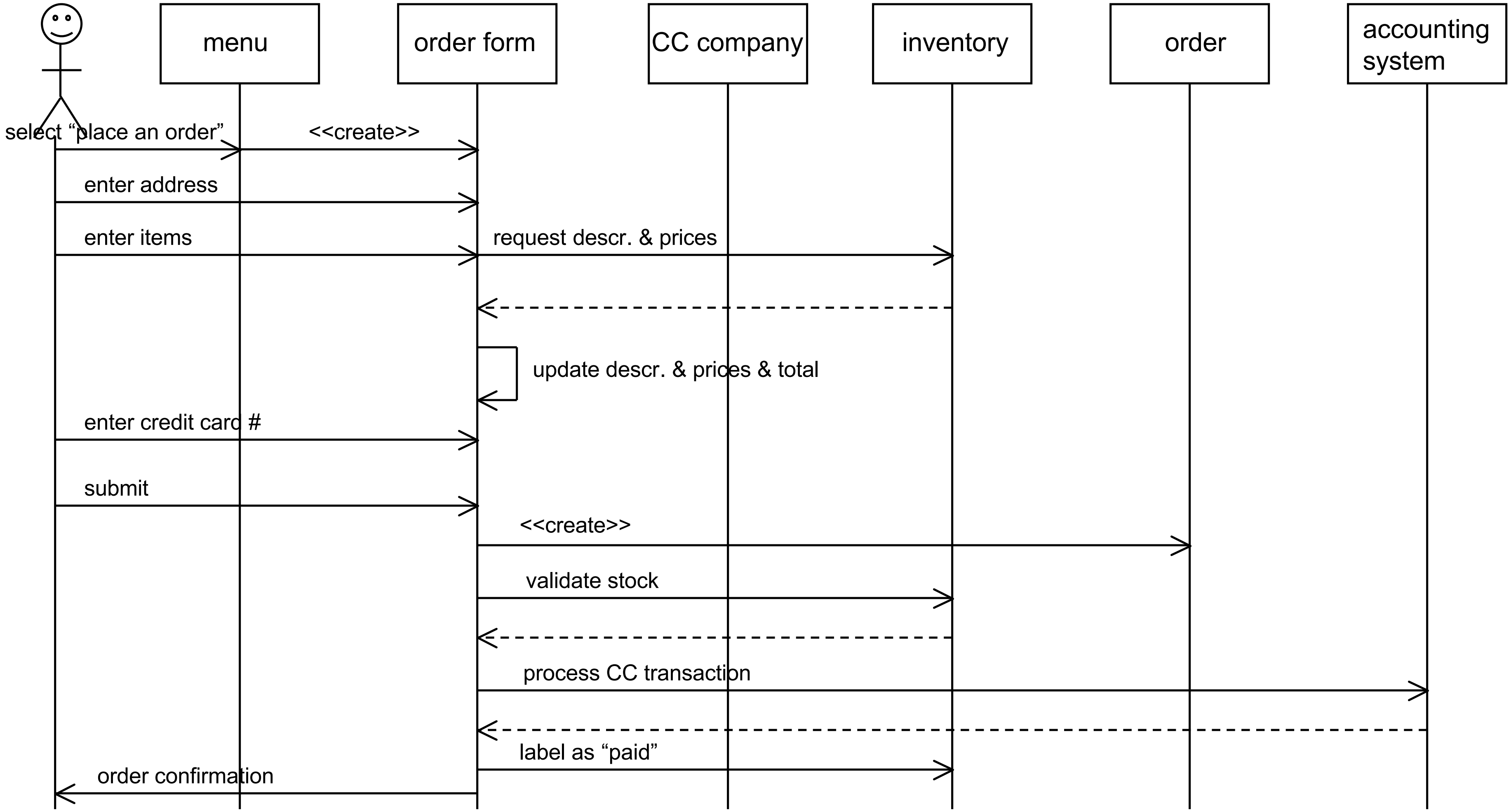
Use Case Diagrams

- Roles
  - Model the context of the system. Define what are the actors that are external to the system
  - Model the requirements of the system. Define what the system should do from an external point of view

Order-Processing Use Case Diagram



Order Processing Sequence Diagram


menu
order form
CC company
inventory
order
accounting system
select "place an order"
<<create>>
enter address
enter items
request descr. & prices
update descr. & prices & total
enter credit card #
submit
<<create>>
validate stock
process CC transaction
label as "paid"
order confirmation