

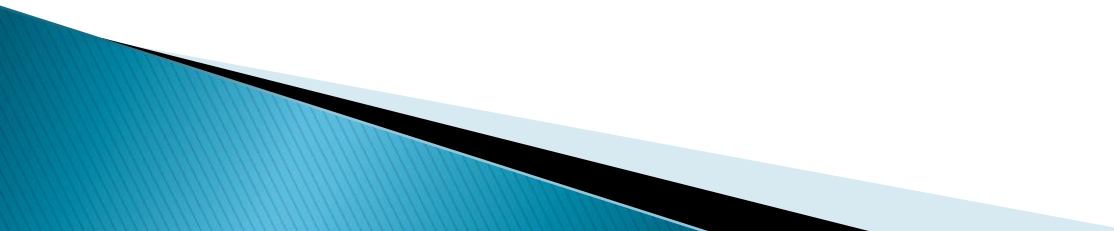
# Introduction to UML

Ana M. Fernández

[fernande@liacs.nl](mailto:fernande@liacs.nl)

Room 150

# Overview

- ▶ What is Modeling?
  - ▶ What is UML?
  - ▶ UML diagrams
  - ▶ UML Modeling tools
- 

# Modelling

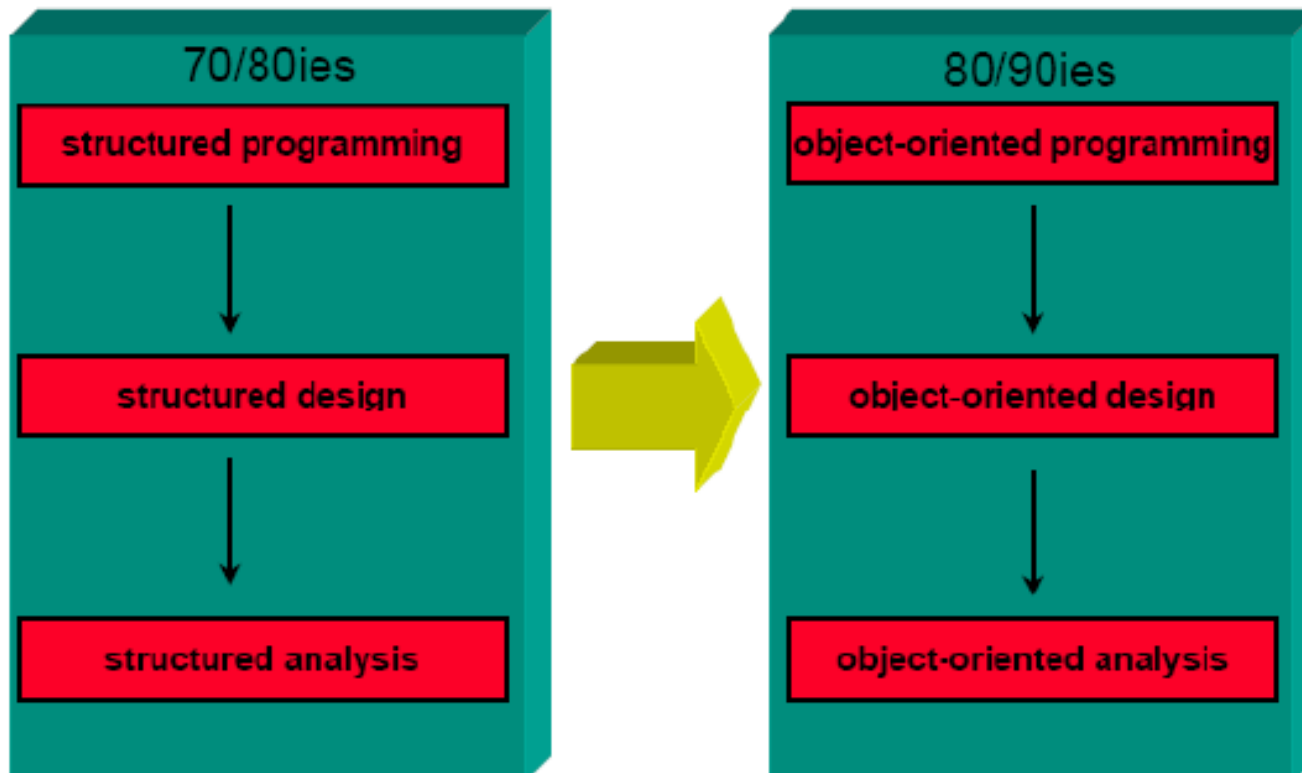
- ▶ *Models are abstractions that portray the essentials of a complex* problem or structure by filtering out nonessential details.
- ▶ Describing a system at a high level of abstraction
  - A model of the system
  - Used for requirements and specifications
- ▶ Models help us organize, visualize, understand, and create complex things.
- ▶ Is it necessary to model software systems?

# Visual Modelling

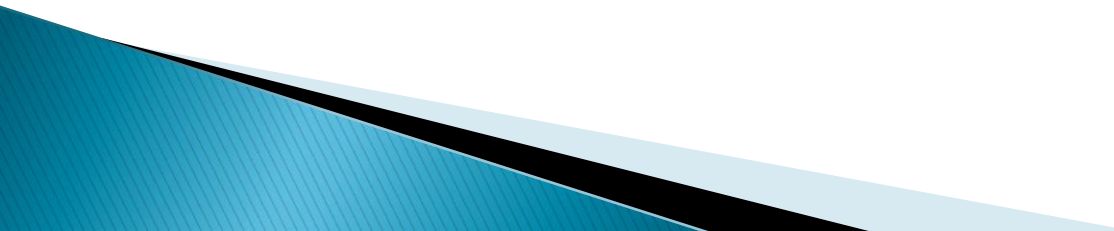
- ▶ *Visual modeling is a way of thinking about problems* using models organized around real-world ideas.
- ▶ Models are useful for
  - Understanding problems
  - Communicating with everyone involved with the project (customer, domain expert, analyst, designers, etc.)
  - Modeling enterprises
  - Preparing documentation
  - Designing programs and databases

# Object Oriented Modelling

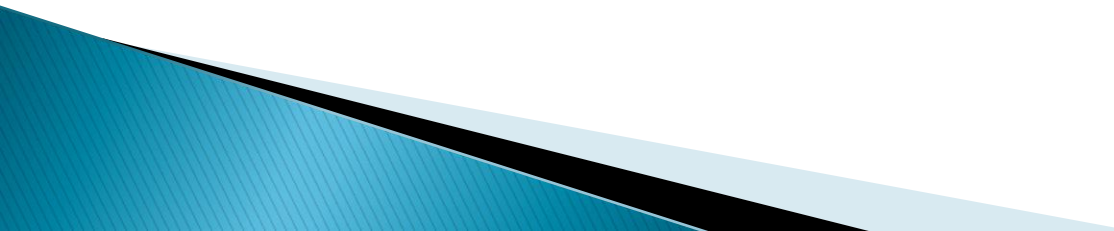
## Evolution of OO Development Methods



# UML

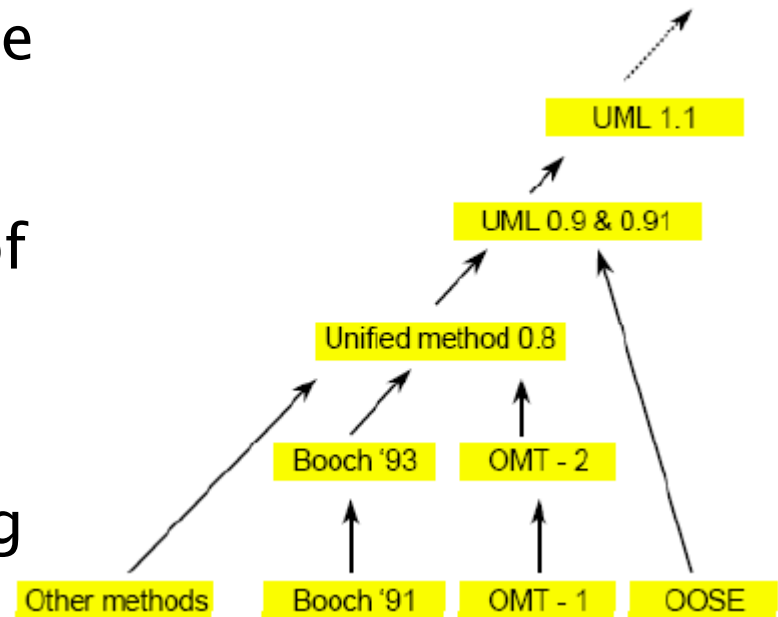
- ▶ UML stands for “Unified Modeling Language”
  - ▶ It is an industry–standard graphical language for specifying, visualizing, constructing, and documenting the artifacts of an object–oriented system under development.
  - ▶ The UML uses mostly graphical notations to express the OO analysis and design of software projects.
  - ▶ Simplifies the complex process of software design
- 

# Why UML?

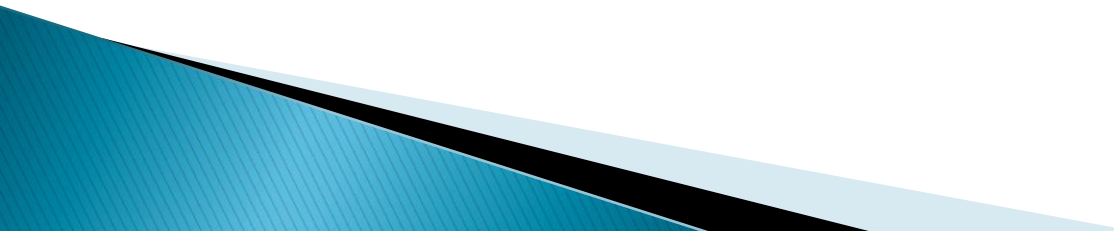
- ▶ Use graphical notation to communicate more clearly than natural language (imprecise) and code(too detailed).
  - ▶ Help acquire an overall view of a system.
  - ▶ UML is *not dependent on any one language or technology*.
  - ▶ UML moves us from fragmentation to standardization.
- 

# The Unified Modelling Language

- ▶ Several different notations for describing object-oriented designs were proposed in the 1980s and 1990s.
- ▶ The Unified Modelling Language is an integration of these notations.
- ▶ It describes notations for a number of different models that may be produced during OO analysis and design.
- ▶ It is now a *de facto standard for OO modelling*.

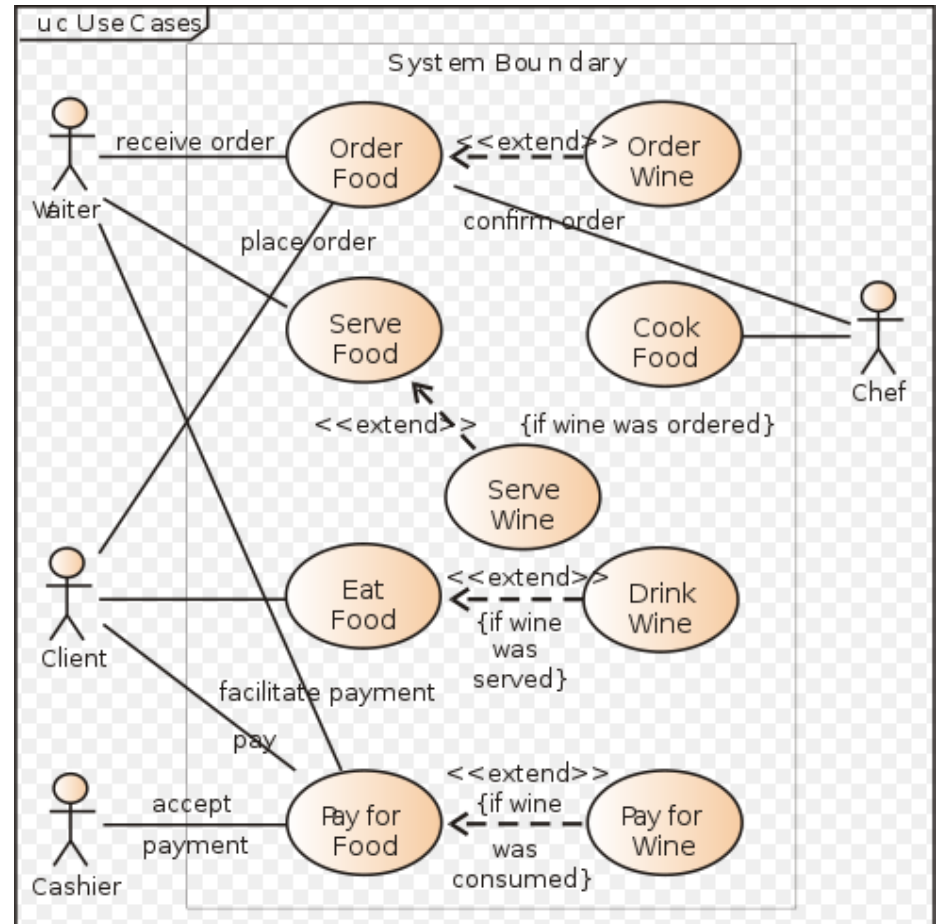


# UML diagrams

- ▶ Use case diagram
  - ▶ Sequence diagram
  - ▶ Collaboration diagram
  - ▶ State diagram
  - ▶ Class diagram
  - ▶ Object diagram
  - ▶ Component diagram
  - ▶ Deployment diagram
  - ▶ Activity diagram
- 

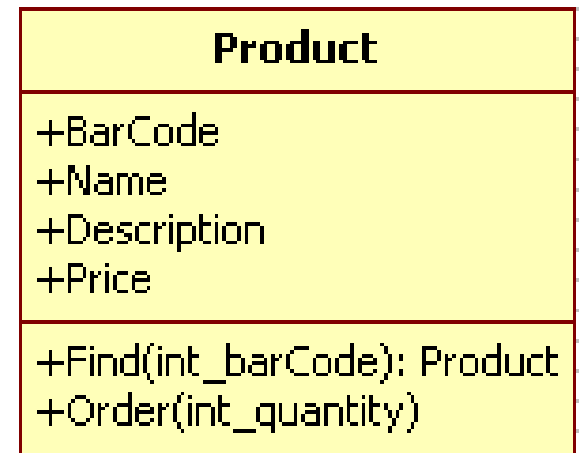
# Use Case diagrams

- ▶ Used for describing a set of user **scenarios**.
- ▶ Mainly used for capturing user requirements.
- ▶ Work like a **contract** between end user and software developers.



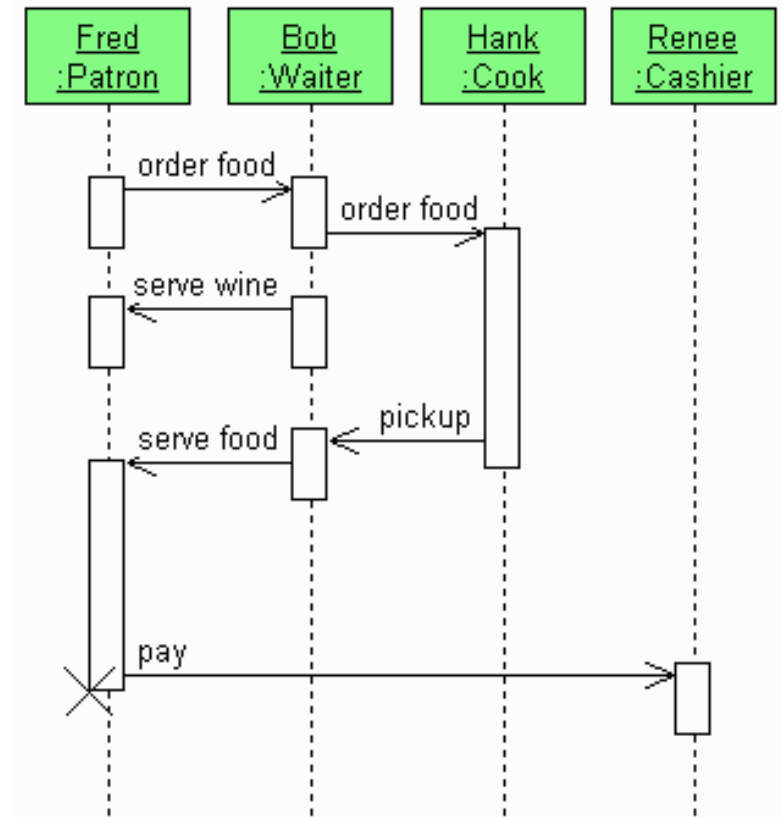
# Class diagrams

- ▶ Used for describing structure and behavior in the use cases
- ▶ Provide a conceptual model of the system in terms of entities and their relationships
- ▶ Used for requirement capture, end-user interaction
- ▶ Detailed class diagrams are used for developers



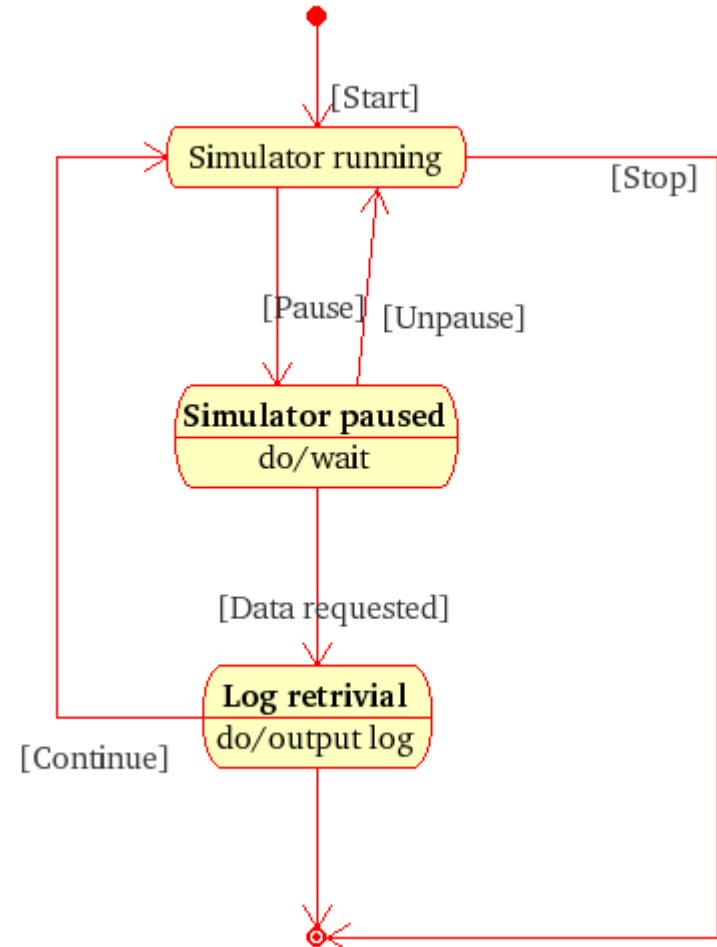
# Sequence diagrams

- ▶ Sequence diagrams demonstrate the behavior of objects in a use case by describing the objects and the messages they pass.
- ▶ The horizontal dimension shows the objects participating in the interaction.
- ▶ The vertical arrangement of messages indicates their order.



# State diagrams

- ▶ State Diagrams show the sequences of states an object goes through during its life cycle in response to stimuli, together with its responses and actions; an abstraction of all possible behaviors.



# UML Modelling Tools

- ▶ Rational Rose ([www.rational.com](http://www.rational.com)) by IBM
- ▶ TogetherSoft Control Center, Borland  
(<http://www.borland.com/together/index.html>)
- ▶ ArgoUML (free software)  
(<http://argouml.tigris.org/>)
- ▶ StarUML (free software)  
<http://staruml.sourceforge.net/en/>
- ▶ Others([http://www.objectsbydesign.com/tools/umltools\\_byCompany.html](http://www.objectsbydesign.com/tools/umltools_byCompany.html) )

# References

- ▶ **UML Distilled: A Brief Guide to the Standard Object Modeling Language** Martin Fowler, Kendall Scott
- ▶ IBM Rational <http://www-306.ibm.com/software/rational/uml/>
- ▶ Practical UML – A Hands–On Introduction for Developers  
[http://www.togethersoft.com/services/practical\\_guides/umlonlinecourse/](http://www.togethersoft.com/services/practical_guides/umlonlinecourse/)
- ▶ **Software Engineering Principles and Practice.** Second Edition; Hans van Vliet.