

Object-Oriented and UML2

Chapter 3



Contents

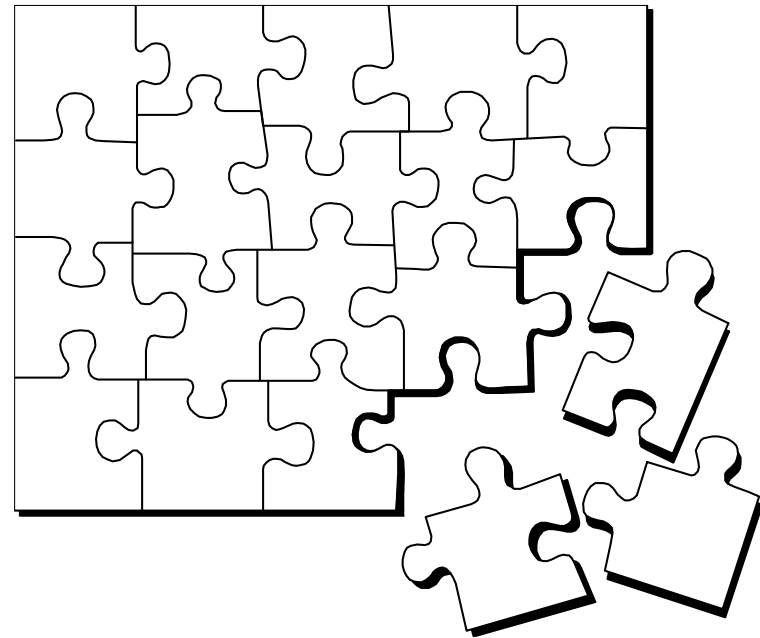
- ❖ **Object Technology**
- ❖ **Visual modeling and UML**
- ❖ **UML diagrams**
 - ◆ **Use case modeling**
 - ◆ **Class diagrams**
 - ◆ **Interaction diagrams**
 - ◆ **Other UML diagrams**

Contents

- ❖ Object Technology
- ❖ Visual modeling and UML
- ❖ UML diagrams
 - ◆ Use case modeling
 - ◆ Class diagrams
 - ◆ Interaction diagrams
 - ◆ Other UML diagrams

What Is Object Technology?

❖ **A set of principles (abstraction, encapsulation, polymorphism) guiding software construction, together with languages, databases, and other tools that support those principles.**

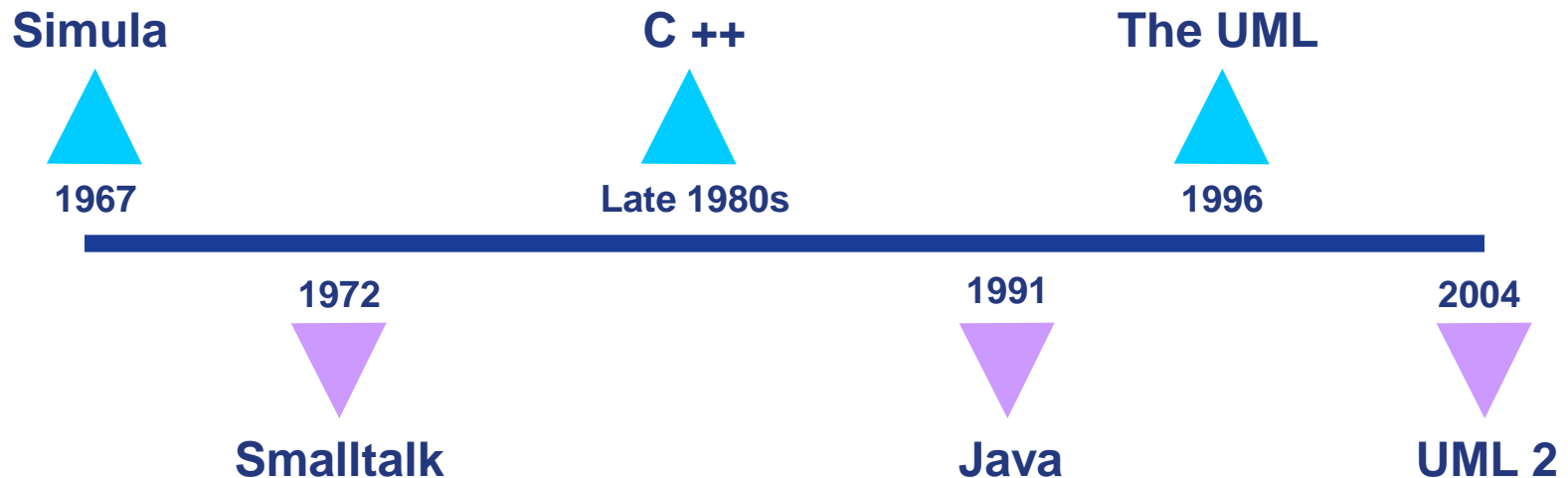


The Strengths of Object Technology

- ❖ Reflects a single paradigm
- ❖ Facilitates architectural and code reuse
- ❖ Reflects real world models more closely
- ❖ Encourages stability
- ❖ Is adaptive to change

The History of Object Technology

❖ Major object technology milestones



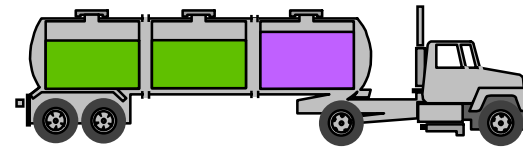
What Is an Object?

❖ Informally, an object represents an entity, either physical, conceptual, or software.

◆ Physical entity

◆ Conceptual entity

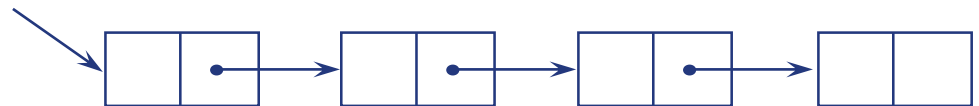
◆ Software entity



Truck



Chemical Process

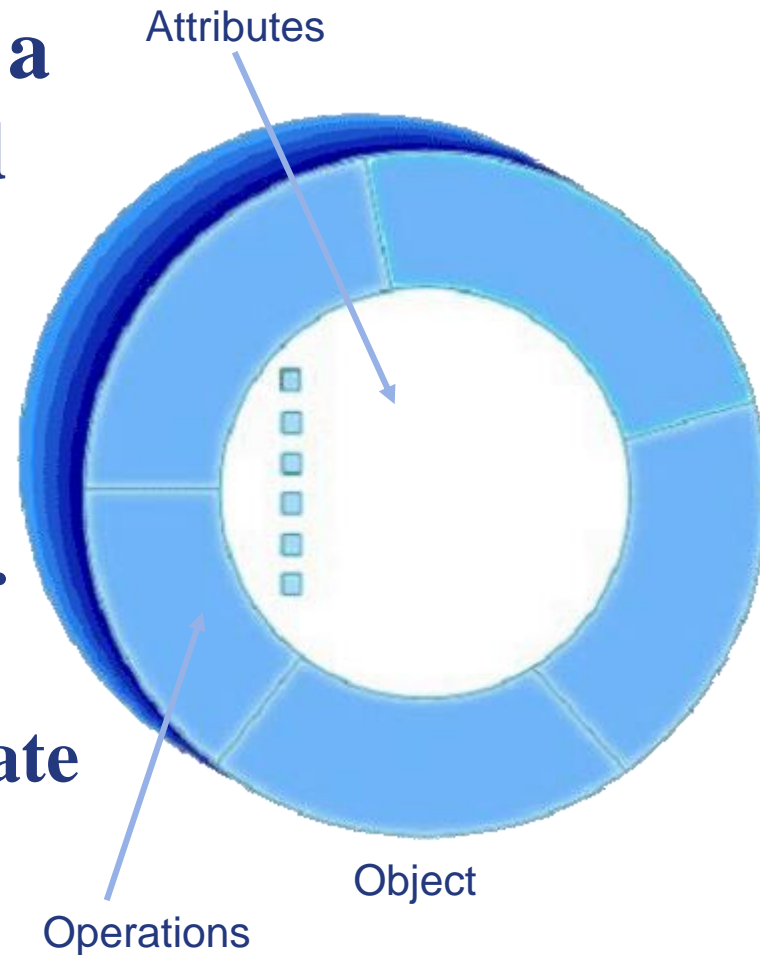


Linked List

A More Formal Definition

❖ An object is an entity with a well-defined boundary and identity that encapsulates state and behavior.

- ◆ State is represented by attributes and relationships.
- ◆ Behavior is represented by operations, methods, and state machines.

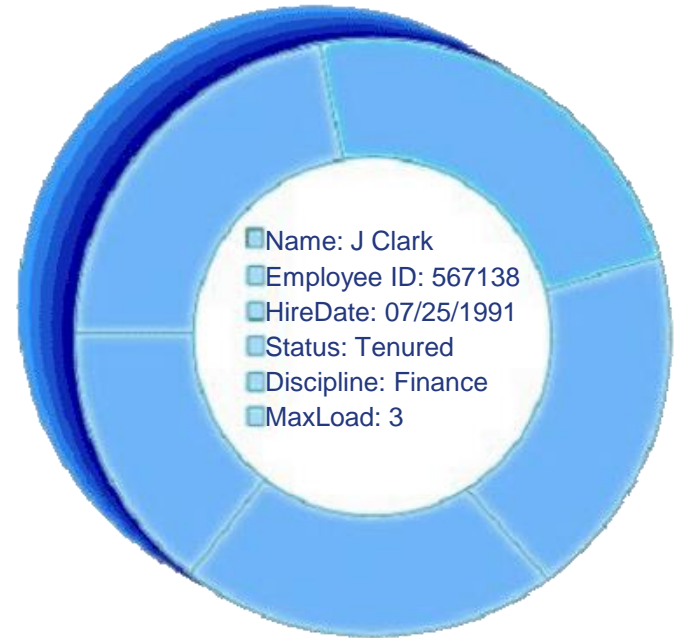


An Object Has State

- ❖ State is a condition or situation during the life of an object, which satisfies some condition, performs some activity, or waits for some event.
- ❖ The state of an object normally changes over time.



Name: J Clark
Employee ID: 567138
Date Hired: July 25, 1991
Status: Tenured
Discipline: Finance
Maximum Course Load: 3 classes



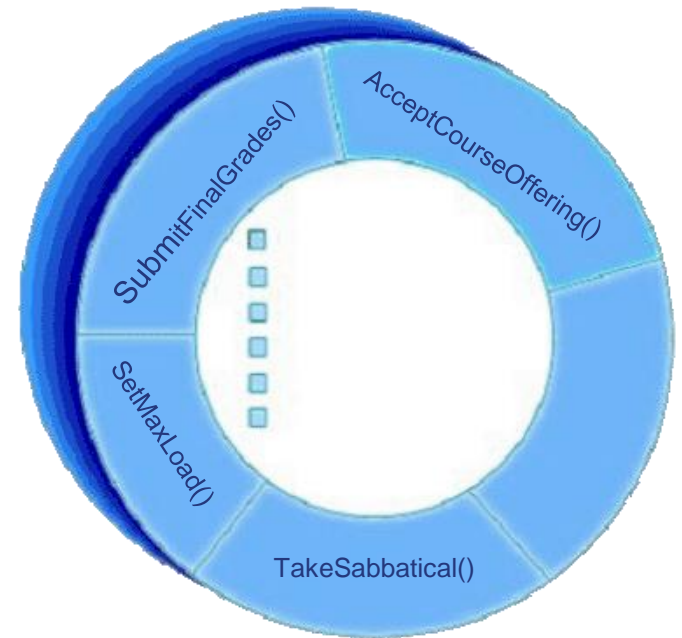
Professor Clark

An Object Has Behavior

- ❖ Behavior determines how an object acts and reacts.
- ❖ The visible behavior of an object is modeled by a set of messages it can respond to (operations that the object can perform).



Professor Clark's behavior
Submit Final Grades
Accept Course Offering
Take Sabbatical
Set Max Load



Professor Clark

Representing Objects in the UML

- ❖ An object is represented as a rectangle with an underlined name.



Professor J Clark

J Clark :
Professor

Named Object

: Professor

Anonymous Object

What Is a Class?

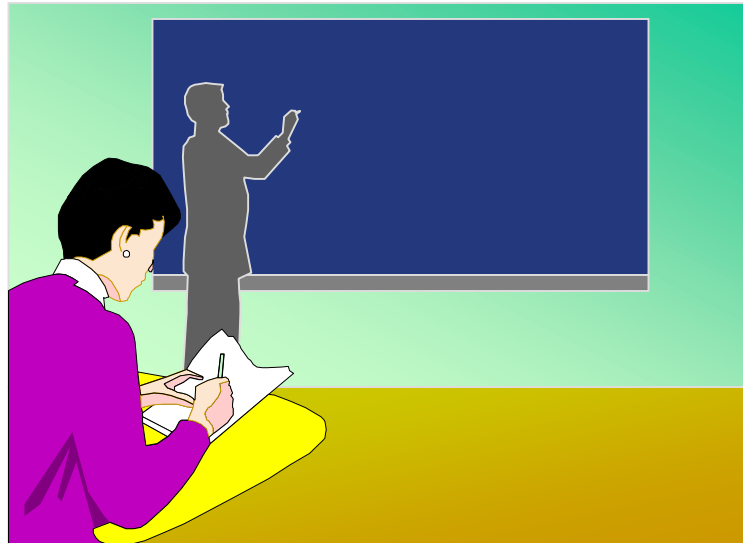
- ❖ A class is a description of a set of objects that share the same attributes, operations, relationships, and semantics.
 - ◆ An object is an instance of a class.
- ❖ A class is an abstraction in that it
 - ◆ Emphasizes relevant characteristics.
 - ◆ Suppresses other characteristics.

A Sample Class

Class Course

Properties

Name
Location
Days offered
Credit hours
Start time
End time



Behavior

Add a student
Delete a student
Get course roster
Determine if it is full

类的抽象，取决于项目上下文



- A. 人
- B. 男人、女人
- C. 老板、员工
- D. 老师、学生

- ❖ A class is an abstraction in that it
 - ◆ Emphasizes **relevant** characteristics.
 - ◆ Suppresses other characteristics.

与需求相关

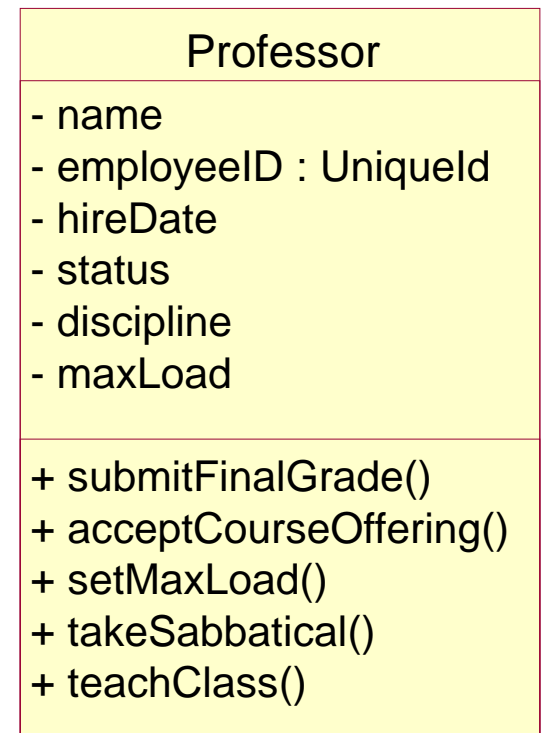
Representing Classes in the UML

❖ A class is represented using a rectangle with three compartments:

◆ The class name

◆ The structure (attributes)

◆ The behavior (operations)

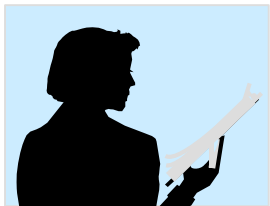


The Relationship between Classes and Objects

- ❖ A class is an abstract definition of an object.
 - ◆ It defines the structure and behavior of each object in the class.
 - ◆ It serves as a template for creating objects.
- ❖ Classes are not collections of objects.



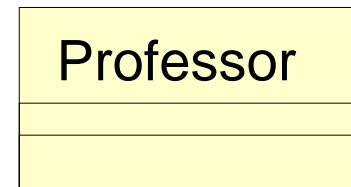
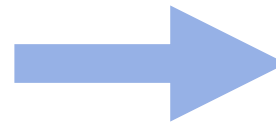
Professor Torpie



Professor Meijer

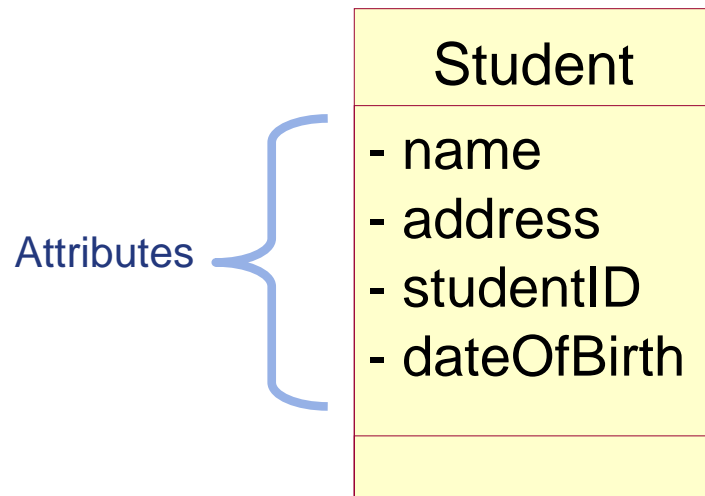


Professor Allen

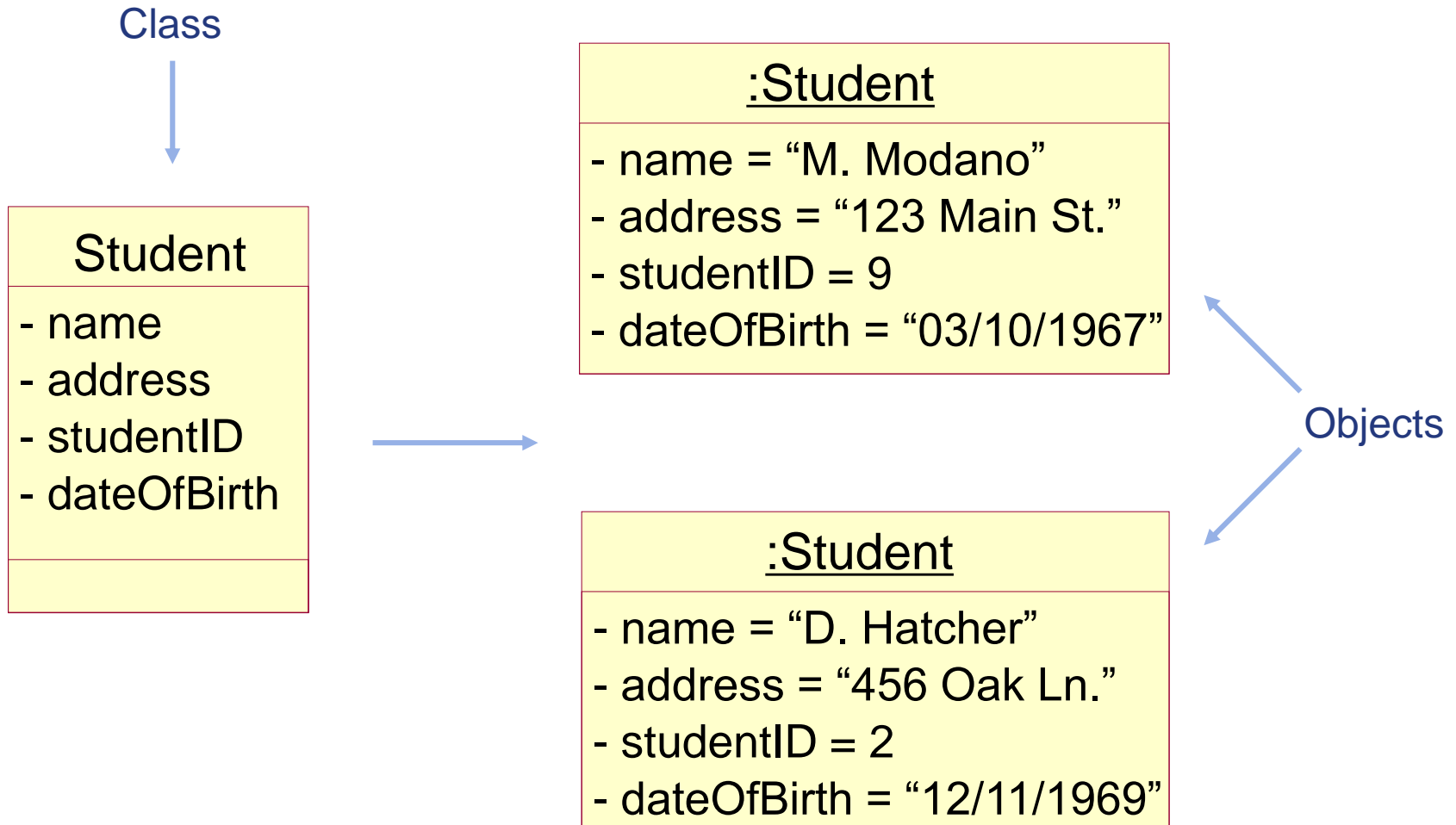


What Is an Attribute?

- ❖ An attribute is a named property of a class that describes the range of values that instances of the property may hold.
 - ◆ A class may have any number of attributes or no attributes at all.

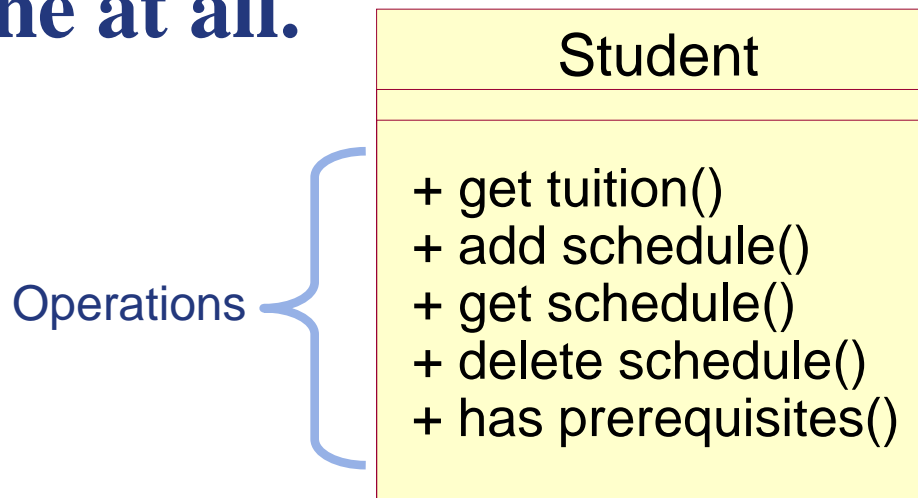


Attributes in Classes and Objects



What Is an Operation?

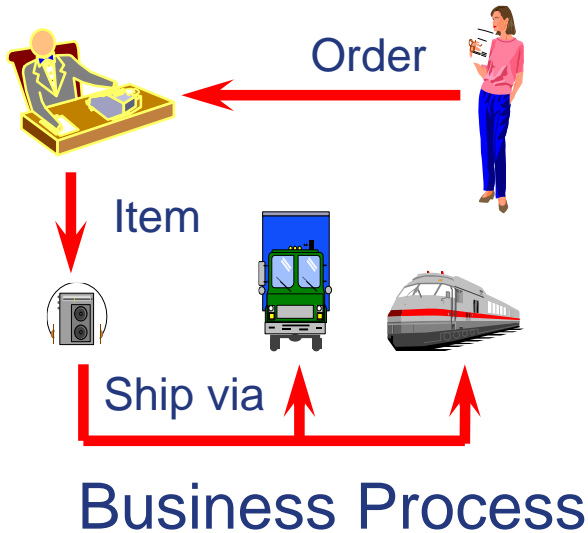
- ❖ A service that can be requested from an object to effect behavior.
- ❖ An operation has a signature, which may restrict the actual parameters that are possible.
- ❖ A class may have any number of operations or none at all.



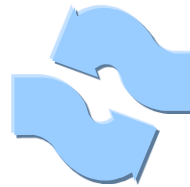
Contents

- ❖ Object Technology
- ❖ Visual modeling and UML
- ❖ UML diagrams
 - ◆ Use case modeling
 - ◆ Class diagrams
 - ◆ Interaction diagrams
 - ◆ Other UML diagrams

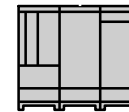
What Is Visual Modeling?



“Modeling captures essential parts of the system.”
Dr. James Rumbaugh



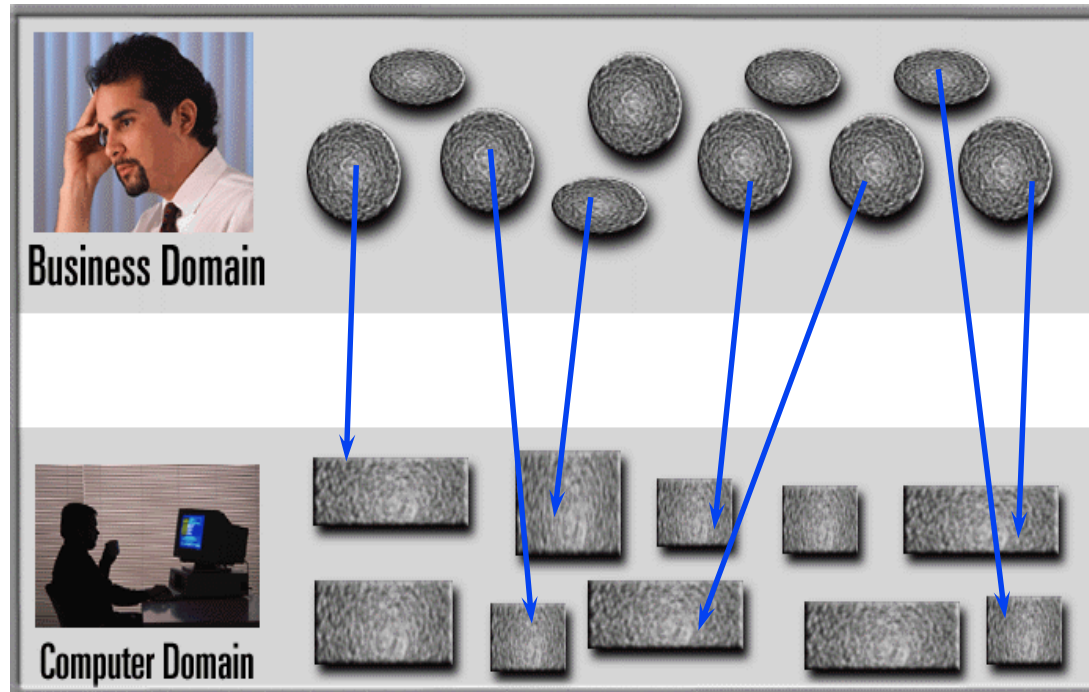
Visual Modeling is modeling using standard graphical notations



Computer System

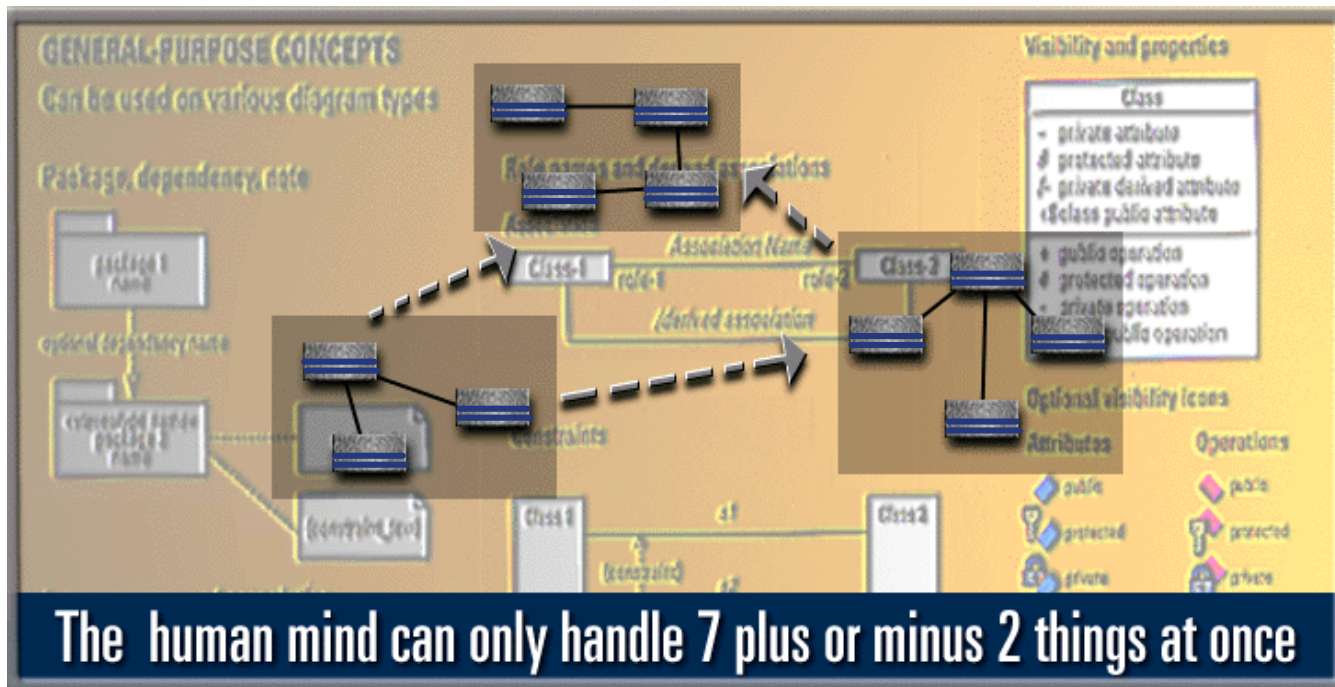
Visual Modeling Is a Communication Tool

Use visual modeling to capture business objects and logic.



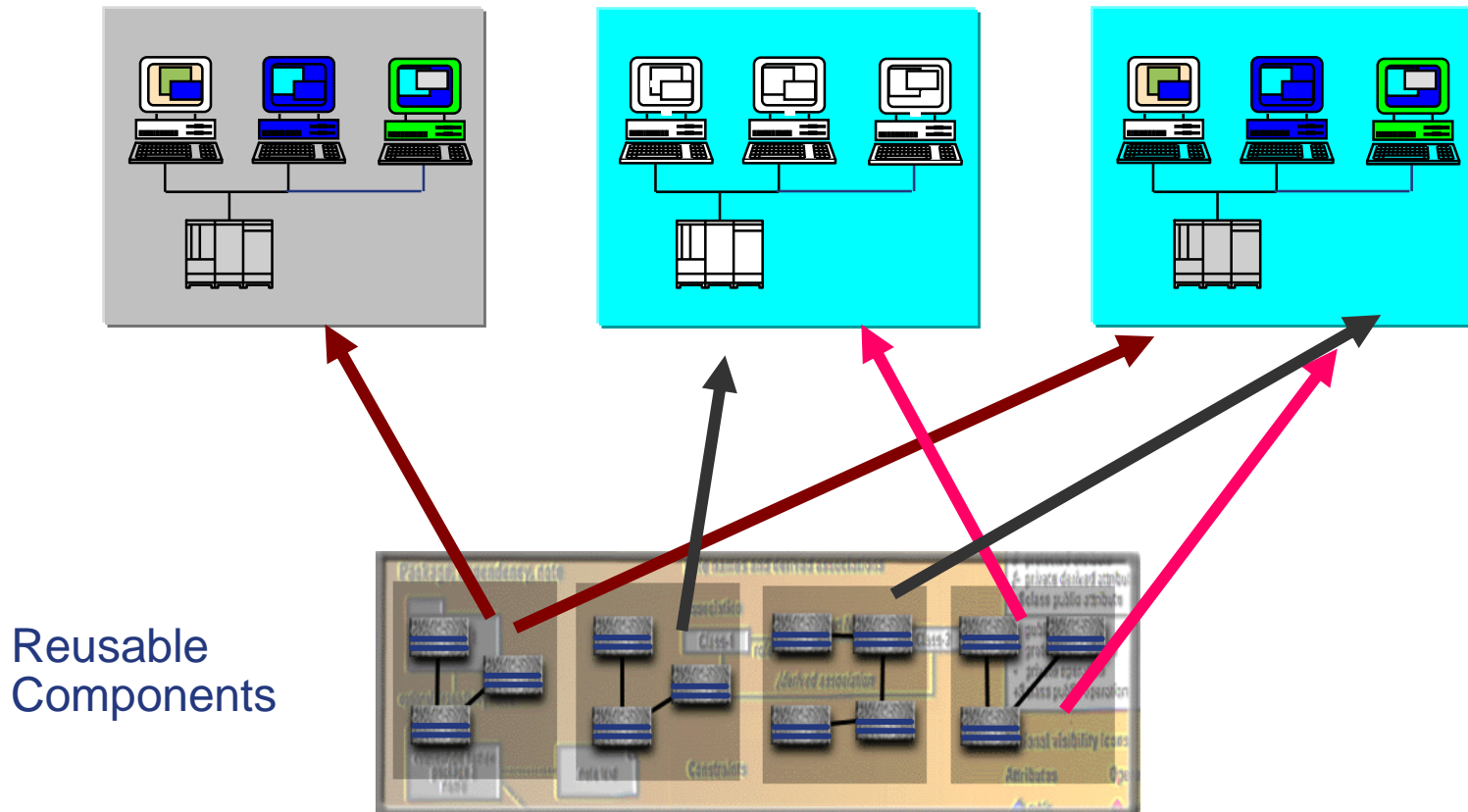
Use visual modeling to analyze and design your application.

Visual Modeling Manages Complexity



Visual Modeling Promotes Reuse

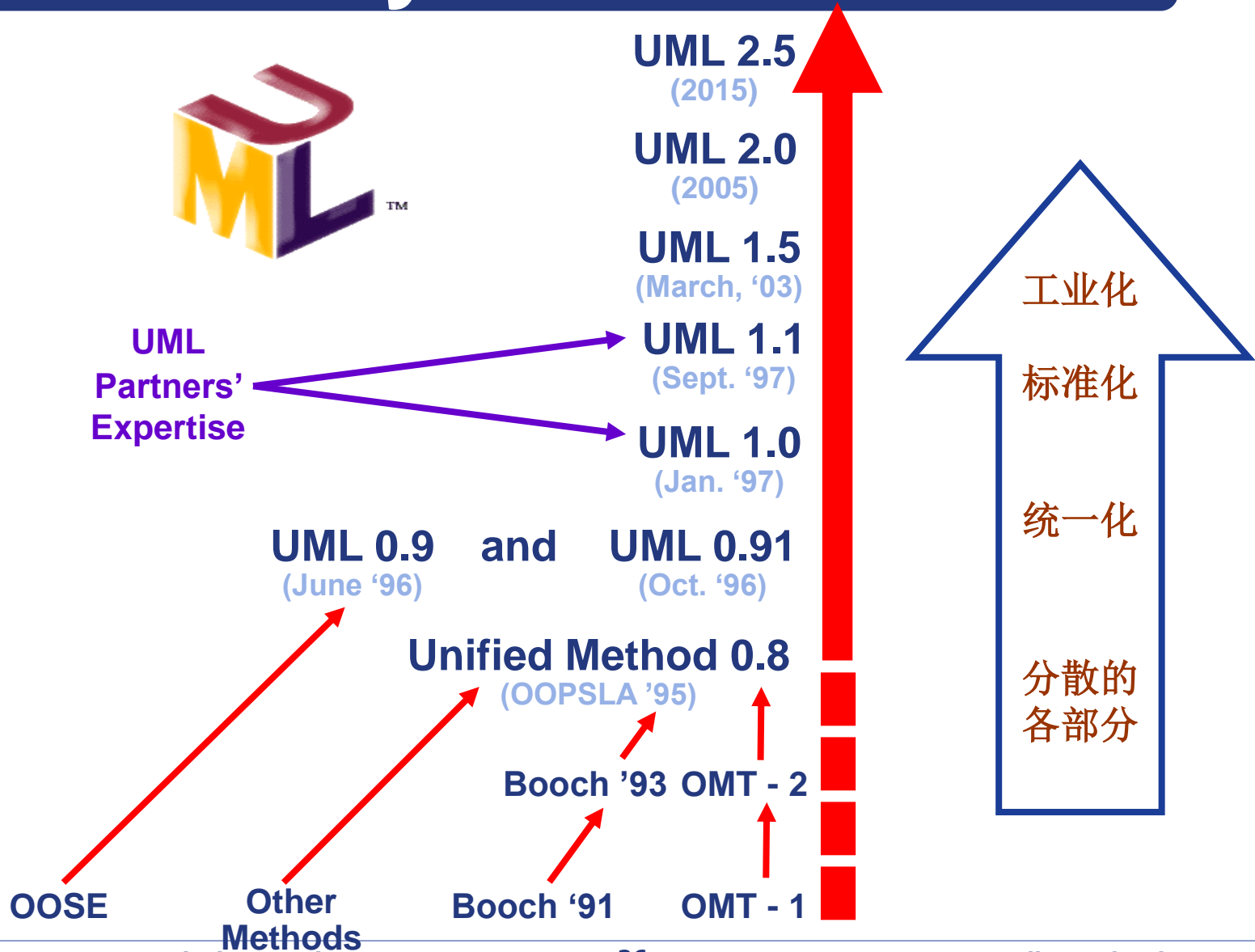
Multiple Systems



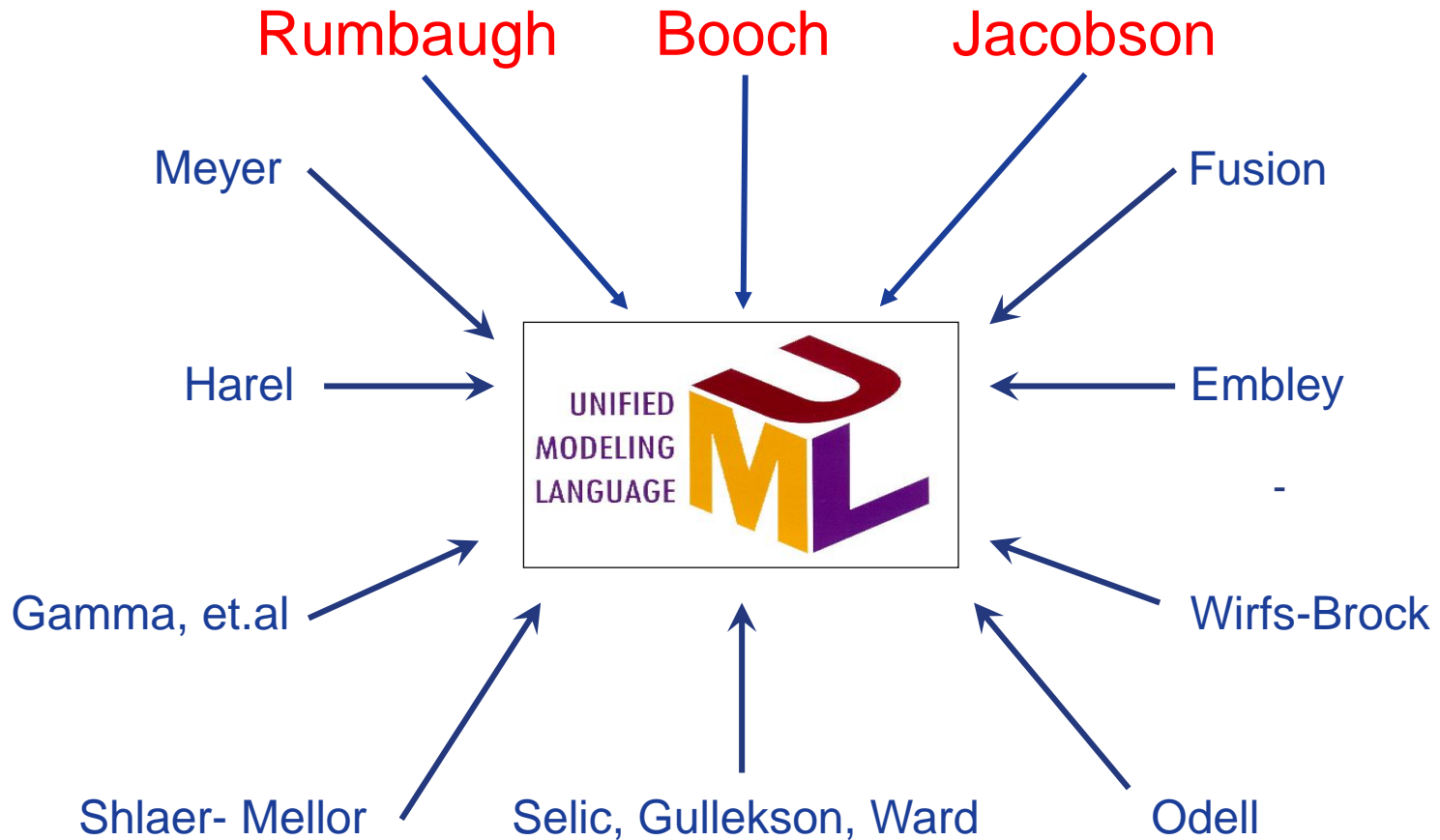
the Unified Modeling Language

- ❖ The **UML** is the standard language for *visualizing*, *specifying*, *constructing*, and *documenting* the artifacts of a software-intensive system.
- ❖ The UML combines the best from
 - ◆ Data modeling
 - ◆ Business modeling
 - ◆ Object modeling
 - ◆ Component modeling

History of the UML



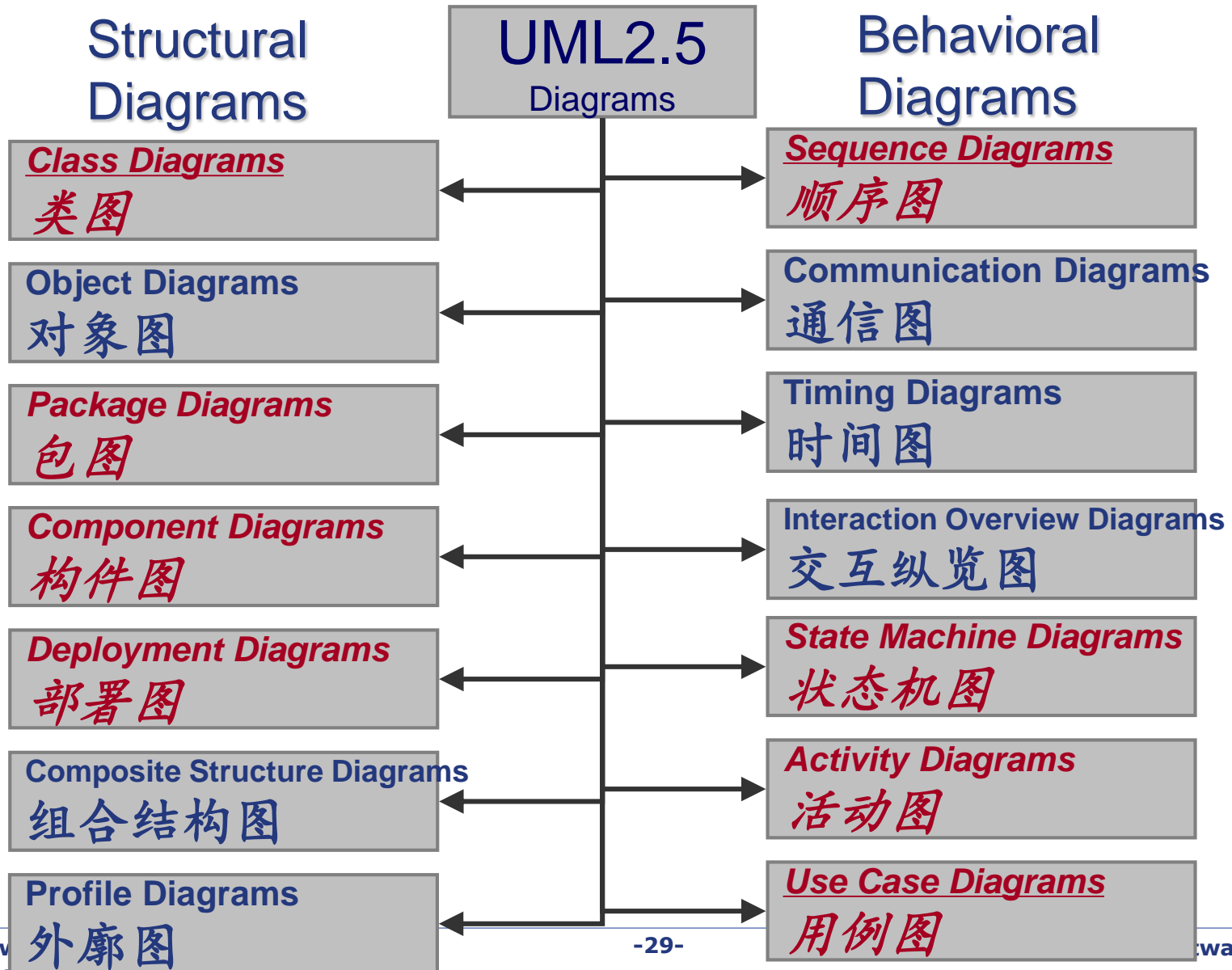
Inputs to the UML



Contents

- ❖ Object Technology
- ❖ Visual modeling and UML
- ❖ UML diagrams
 - ◆ Use case modeling
 - ◆ Class diagrams
 - ◆ Interaction diagrams
 - ◆ Other UML diagrams

The Diagrams in UML 2.5



❖ IBM Rational Suite

◆ Rational Rose 2003

- 经典的UML建模工具，目前仍有广泛的应用
- 不支持UML2.0

◆ Rational Software Architecture

- IBM兼并Rational之后，重新基于Eclipse平台构建的集成开发平台，提供从业务建模、需求分析、设计到系统实现的完整环境

◆ IBM Rational Rhapsody

- IBM兼并另一家UML建模工具后重新发布的产品
- 主要用于嵌入式领域建模，涉及软硬件等各个层次的模型

UML Tools (cont.)

❖ Enterprise Architect

◆ <http://www.sparxsystems.com.au/>

❖ PowerDesigner

◆ <http://www.sybase.com/products/powerdesigner/>

❖ MagicDraw

◆ <http://www.magicdraw.com/>

❖ StarUML

◆ <http://staruml.sourceforge.net>

❖ UModel

◆ http://www.altova.com/products/umodel/uml_tool.html

❖ Kant&Plato

◆ 楚凡科技（中国） <http://www.trufun.net/>

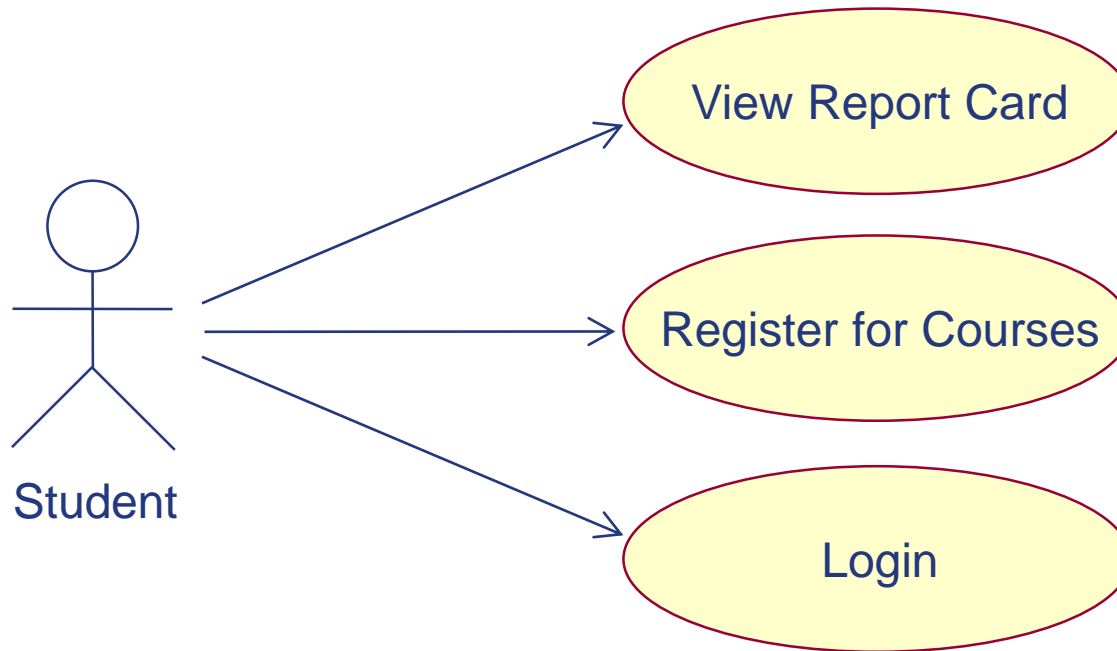
❖ <http://www.umlchina.com/Tools/Newindex1.htm>

Contents

- ❖ Object Technology
- ❖ Principles of Object-Oriented
- ❖ Visual modeling and UML
- ❖ UML diagrams
 - ◆ Use case modeling
 - ◆ Class diagrams
 - ◆ Interaction diagrams
 - ◆ Other UML diagrams

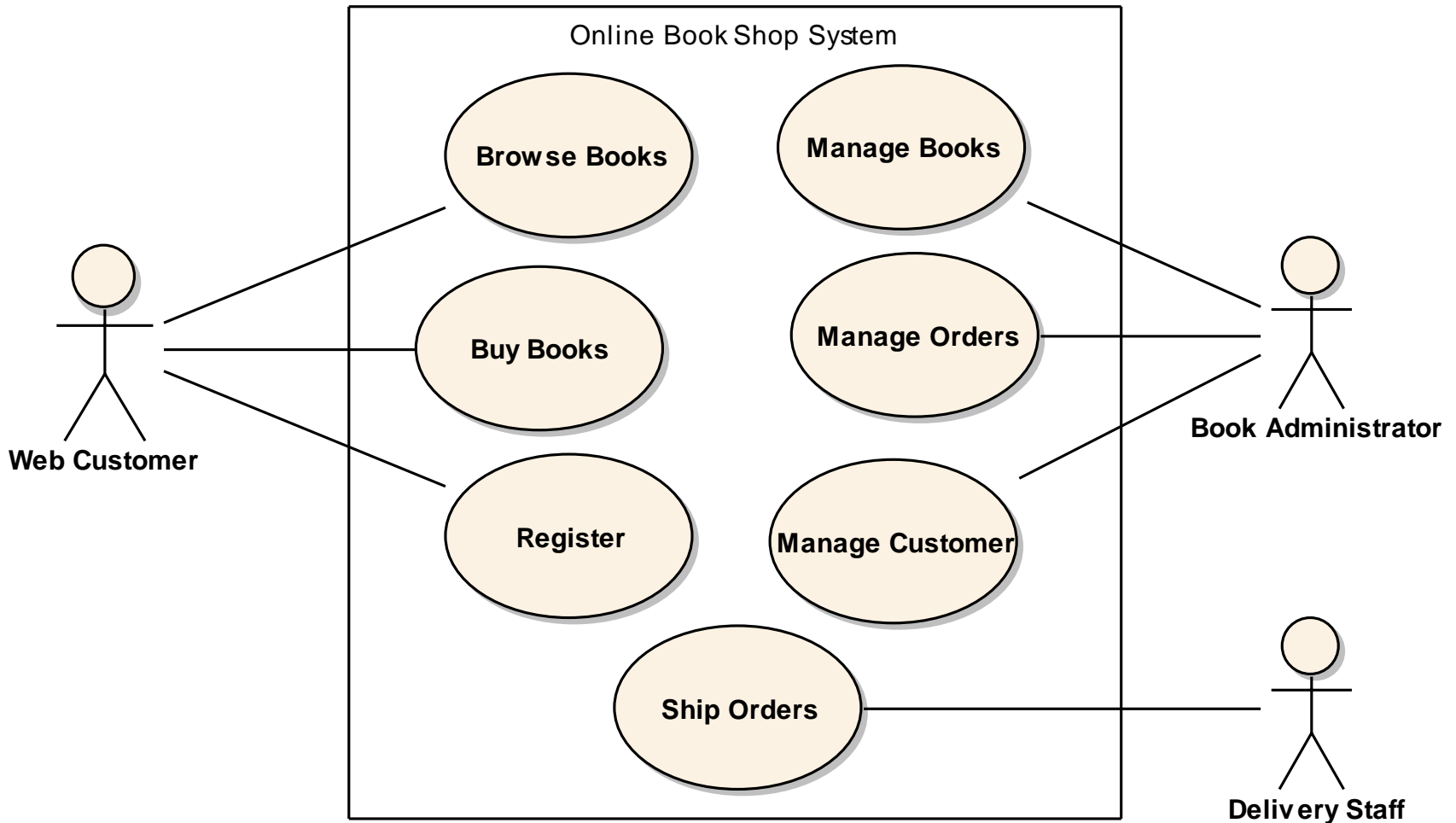
What Is a Use-Case Model?

- ❖ A model that describes a system's functional requirements in terms of use cases.
- ❖ A model of the system's intended functions (use cases) and its environment (actors).



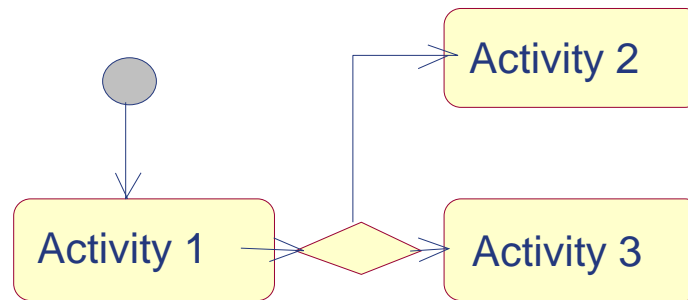
OnlineBookShop Use Case Model

uc Primary Use Cases



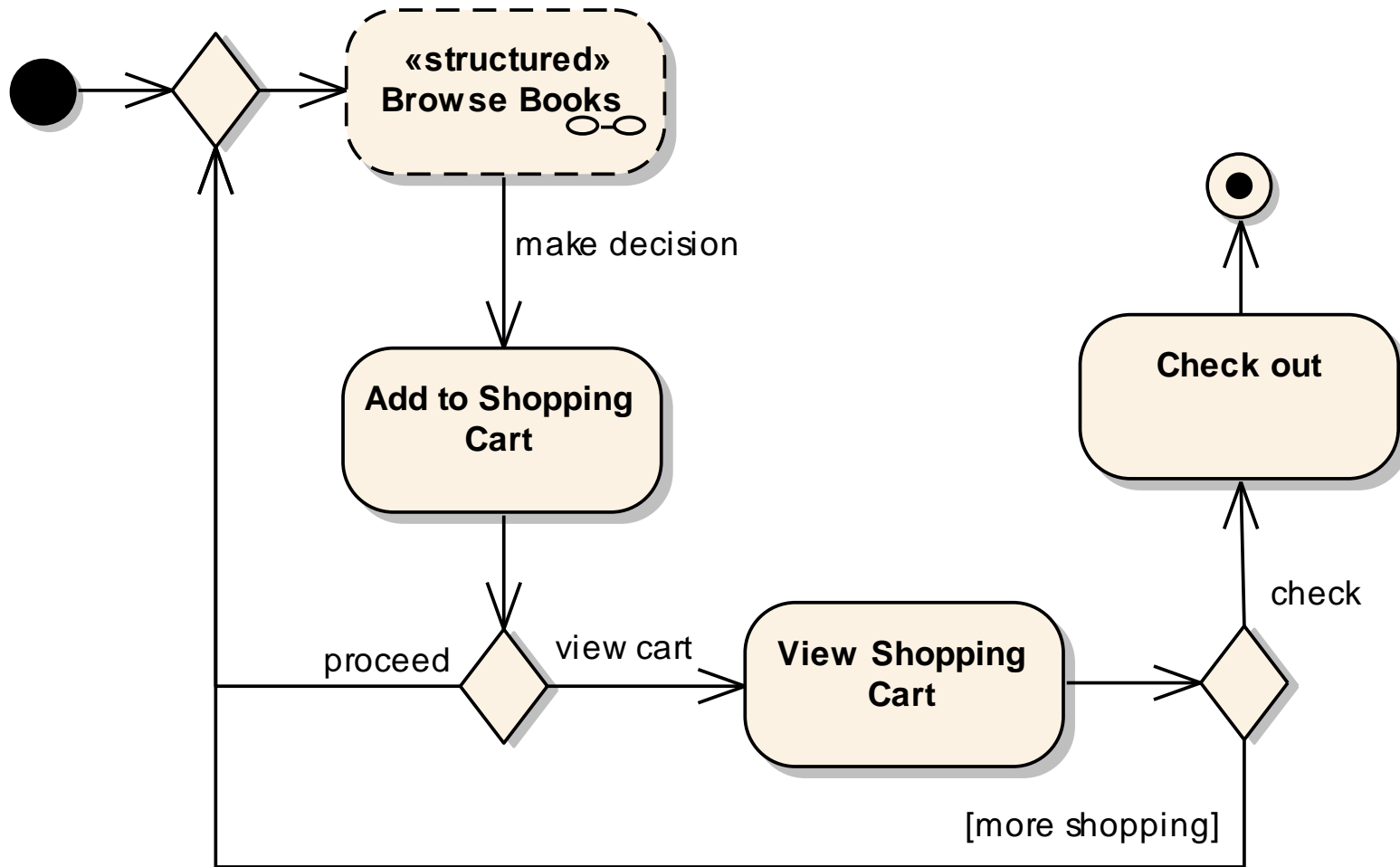
What Is an Activity Diagram?

- ❖ An activity diagram in the use-case model can be used to capture the activities and actions performed in a use case.
- ❖ It is essentially a flow chart, showing flow of control from one activity or action to another.



Example: Activity Diagram

act Activ ...

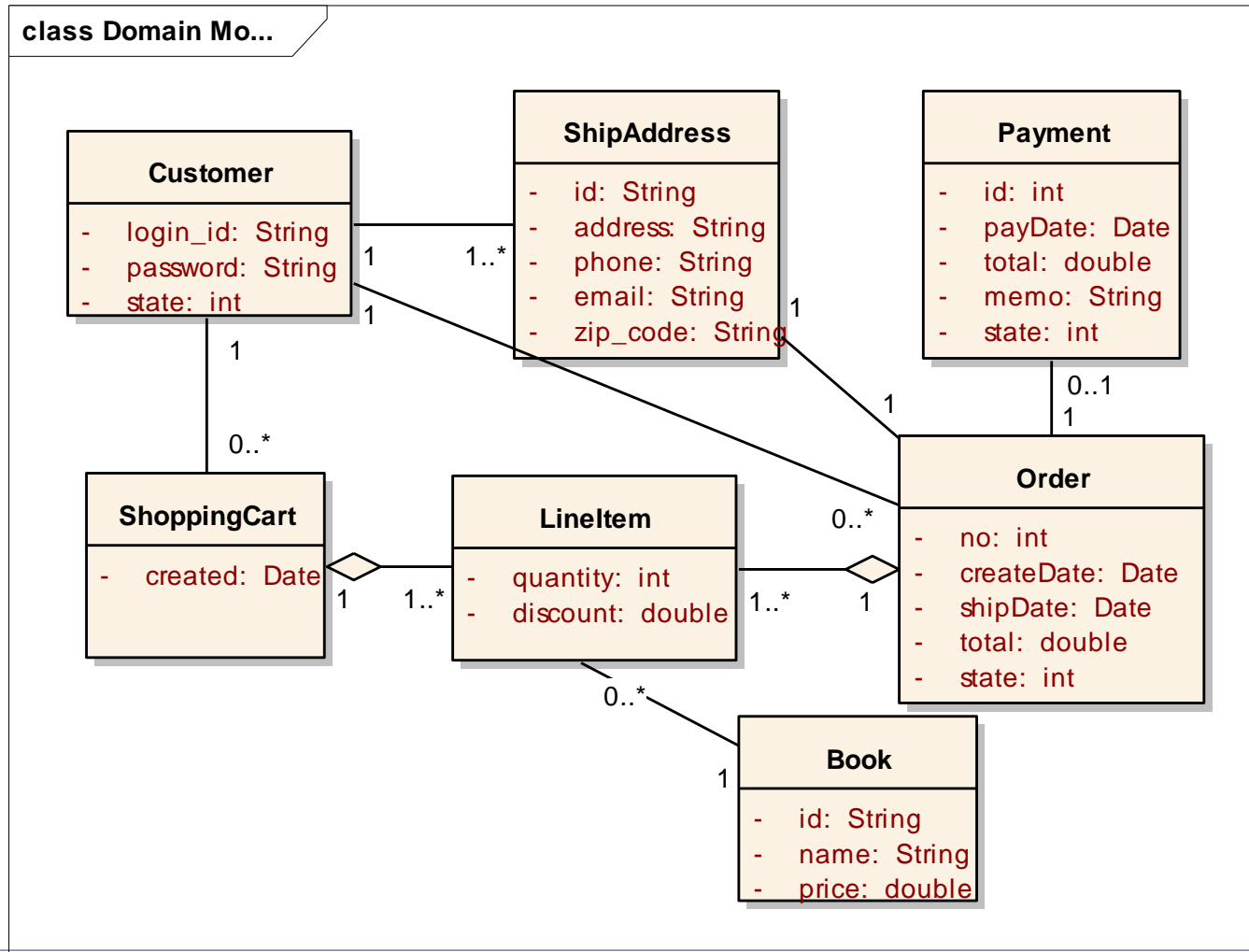


Contents

- ❖ Object Technology
- ❖ Visual modeling and UML
- ❖ UML diagrams
 - ◆ Use case modeling
 - ◆ Class diagrams
 - ◆ Interaction diagrams
 - ◆ Other UML diagrams

What Is a Class Diagram?

❖ Static view of a system



Class Relationships

❖ Association

- ◆ The semantic relationship between two or more classifiers that specifies connections among their instances.
- ◆ A structural relationship specifying that objects of one thing are connected to objects of another thing.

❖ Aggregation

- ◆ A special form of association that models a whole-part relationship between the aggregate (the whole) and its parts.
- ◆ An aggregation is an “is a part-of” relationship.

❖ Generalization

Contents

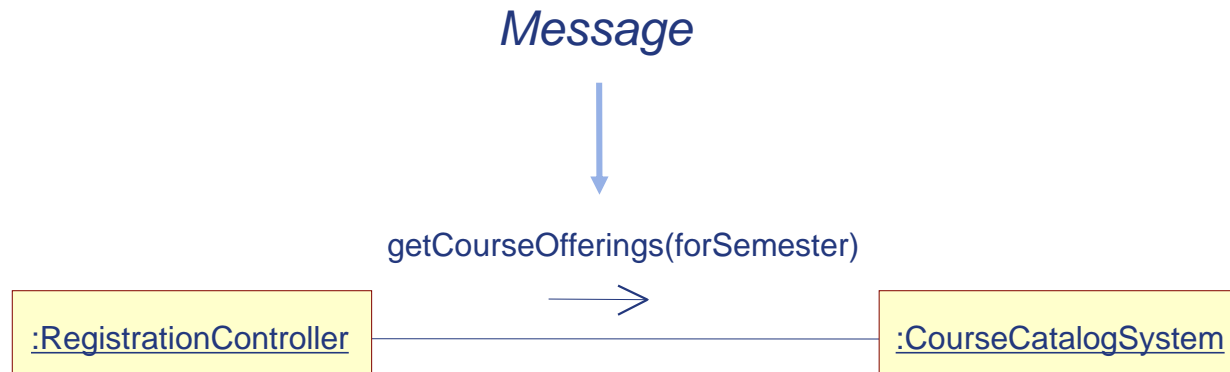
- ❖ Object Technology
- ❖ Visual modeling and UML
- ❖ UML diagrams
 - ◆ Use case modeling
 - ◆ Class diagrams
 - ◆ Interaction diagrams
 - ◆ Other UML diagrams

Objects Need to Collaborate

- ❖ **Objects are useless unless they can collaborate to solve a problem.**
 - ◆ **Each object is responsible for its own behavior and status.**
 - ◆ **No one object can carry out every responsibility on its own.**
- ❖ **How do objects interact with each other?**
 - ◆ **They interact through messages.**

Objects Interact with Messages

- ❖ A message shows how one object asks another object to perform some activity.



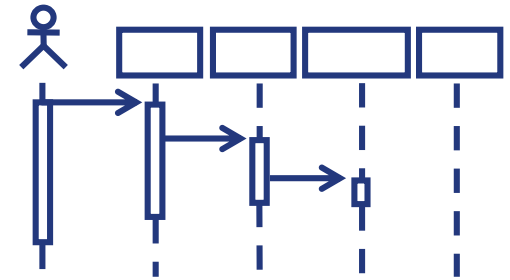
What is an Interaction Diagram?

- ❖ **Generic term that applies to several diagrams that emphasize object interactions**
 - ◆ **Sequence Diagram**
 - ◆ **Communication Diagram**
- ❖ **Specialized Variants**
 - ◆ **Timing Diagram**
 - ◆ **Interaction Overview Diagram**

Interaction Diagrams

❖ Sequence Diagram

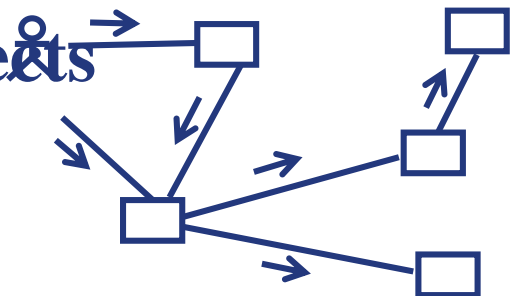
- ◆ Time oriented view of object interaction



Sequence Diagrams

❖ Communication Diagram

- ◆ Structural view of messaging objects
- ◆ taken from the Collaboration diagram concept of UML1

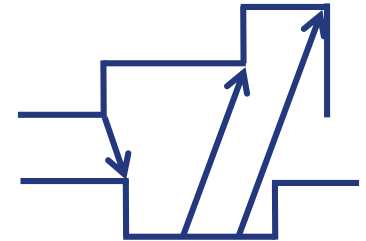


Communication Diagrams

Interaction Diagrams

❖ Timing Diagram

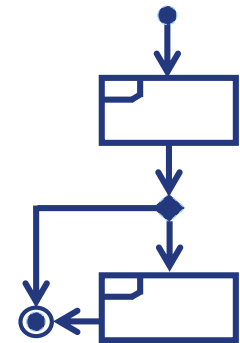
- ◆ Time constraint view of messages involved in an interaction



Timing Diagrams

❖ Interaction Overview Diagram

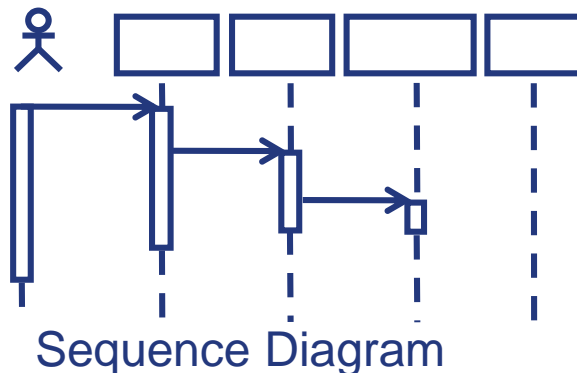
- ◆ High level view of interaction sets combined into logic sequence



Interaction Overview Diagrams

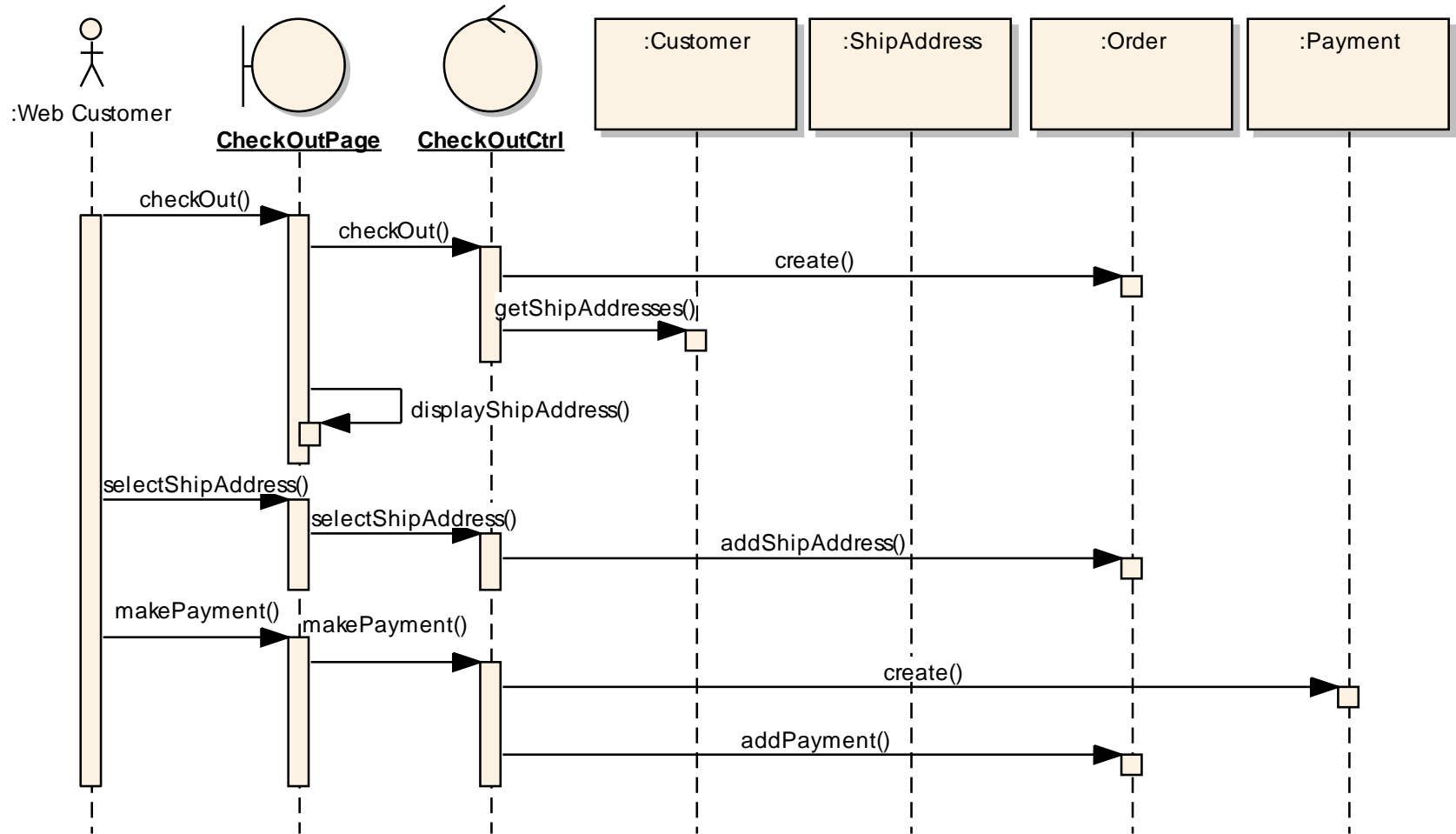
What Is a Sequence Diagram?

- ❖ A sequence diagram is an interaction diagram that emphasizes the time ordering of messages.
- ❖ The diagram shows:
 - ◆ The objects participating in the interaction.
 - ◆ The sequence of messages exchanged.



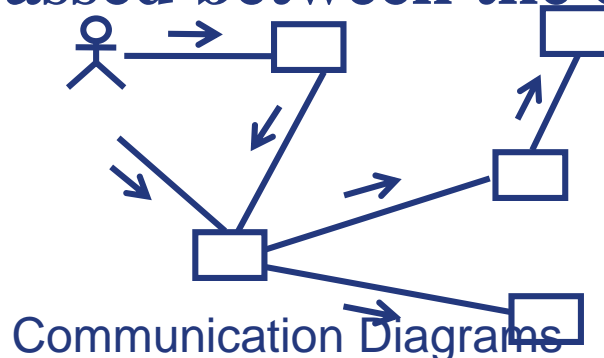
Sequence Diagram: Check out

sd Check out - Realization



What Is a Communication Diagram?

- ❖ A communication diagram emphasizes the organization of the objects that participate in an interaction.
- ❖ The communication diagram shows:
 - ◆ The objects participating in the interaction.
 - ◆ Links between the objects.
 - ◆ Messages passed between the objects.

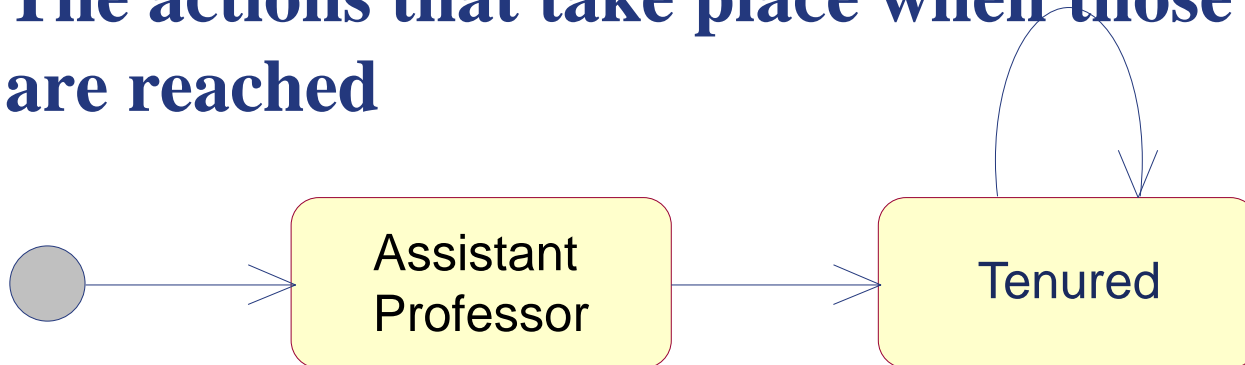


Contents

- ❖ Object Technology
- ❖ Visual modeling and UML
- ❖ UML diagrams
 - ◆ Use case modeling
 - ◆ Class diagrams
 - ◆ Interaction diagrams
 - ◆ Other UML diagrams

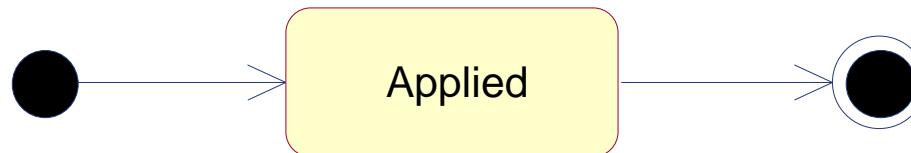
What Are State Machine Diagrams?

- ❖ A state machine diagram models dynamic behavior.
- ❖ It specifies the sequence of states in which an object can exist:
 - ◆ The events and conditions that cause the object to reach those states
 - ◆ The actions that take place when those states are reached



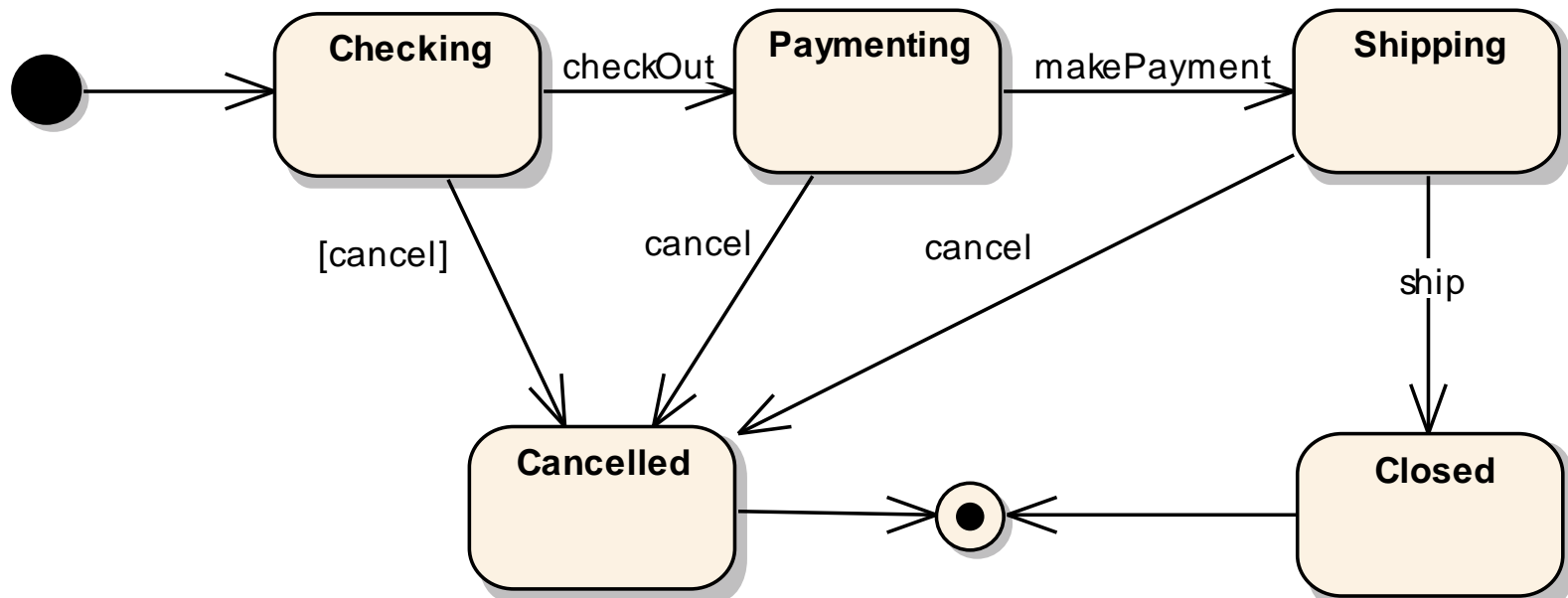
Special States

- ❖ The initial state is the state entered when an object is created.
 - ◆ An initial state is mandatory.
 - ◆ Only one initial state is permitted.
 - ◆ The initial state is represented as a solid circle.
- ❖ A final state indicates the end of life for an object.
 - ◆ A final state is optional.
 - ◆ A final state is indicated by a bull's eye.
 - ◆ More than one final state may exist.



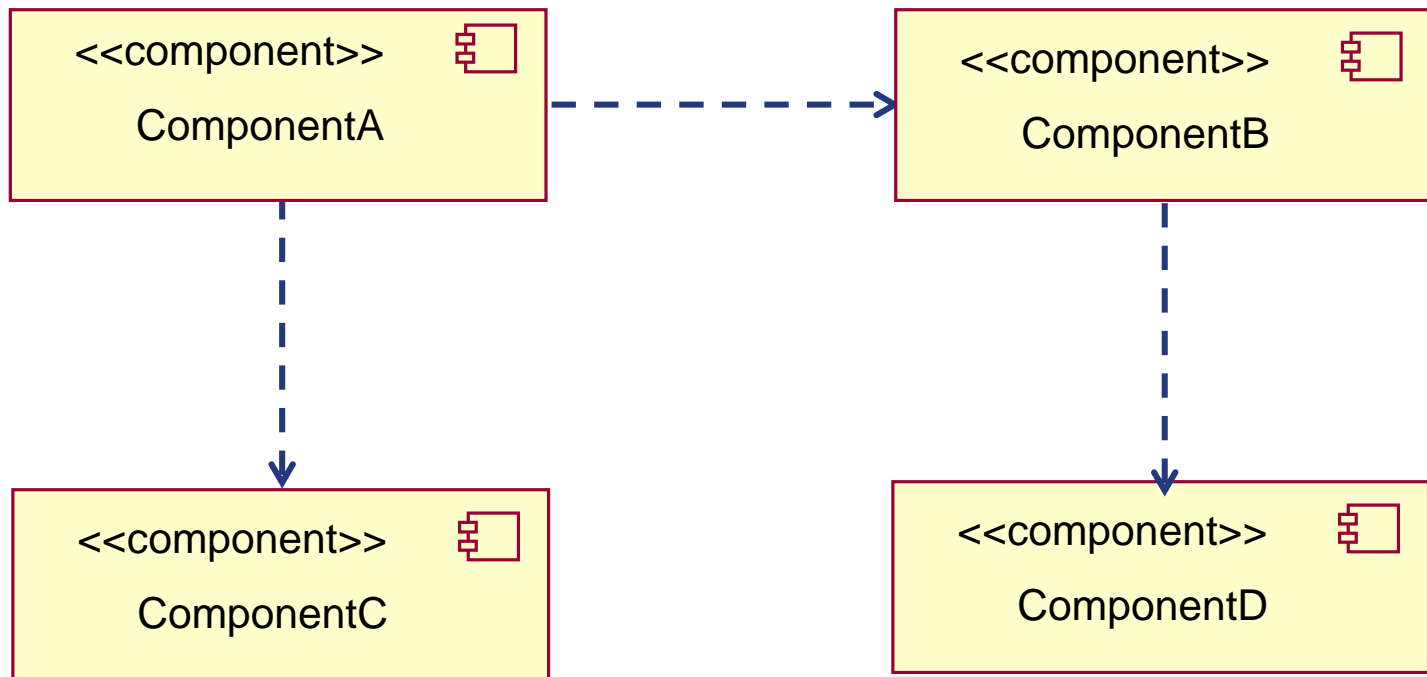
Statemachine Diagram: Order

stm StateMachi...



What Is a Component Diagram?

❖ A diagram that shows the organizations and dependencies among components



What Is a Deployment Diagram?

- ❖ The deployment diagram shows:
 - ◆ Configuration of processing nodes at run-time
 - ◆ Communication links between these nodes
 - ◆ Deployed artifacts that reside on them
- ❖ The deployment diagram consists of one or more:
 - ◆ Nodes: processing elements with at least one processor and memory
 - ◆ Connectors between nodes and/or devices

Deployment Diagram

