

## **UML2 ( UML2.0 )**

**James Rumbaugh, Ivar Jacobson, Grady Booch,  
“The Unified Modeling Language Reference Manual, Second Edition”,  
Addison-Wesley, 2005.**

**Koichiro Ochimizu  
Japan Advanced Institute of  
Science and technologies  
School of Information Science**

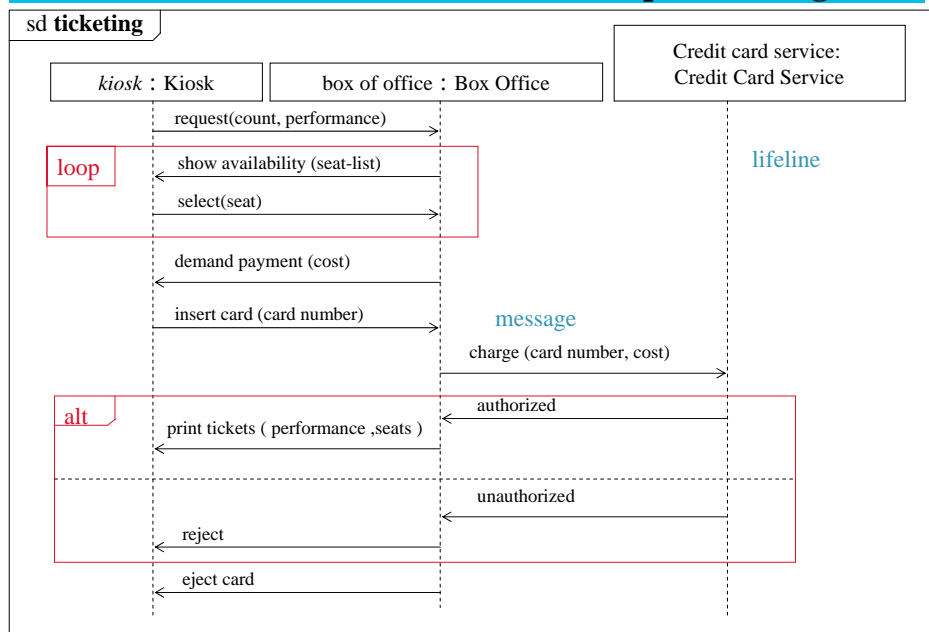
## **Schedule(3/3)**

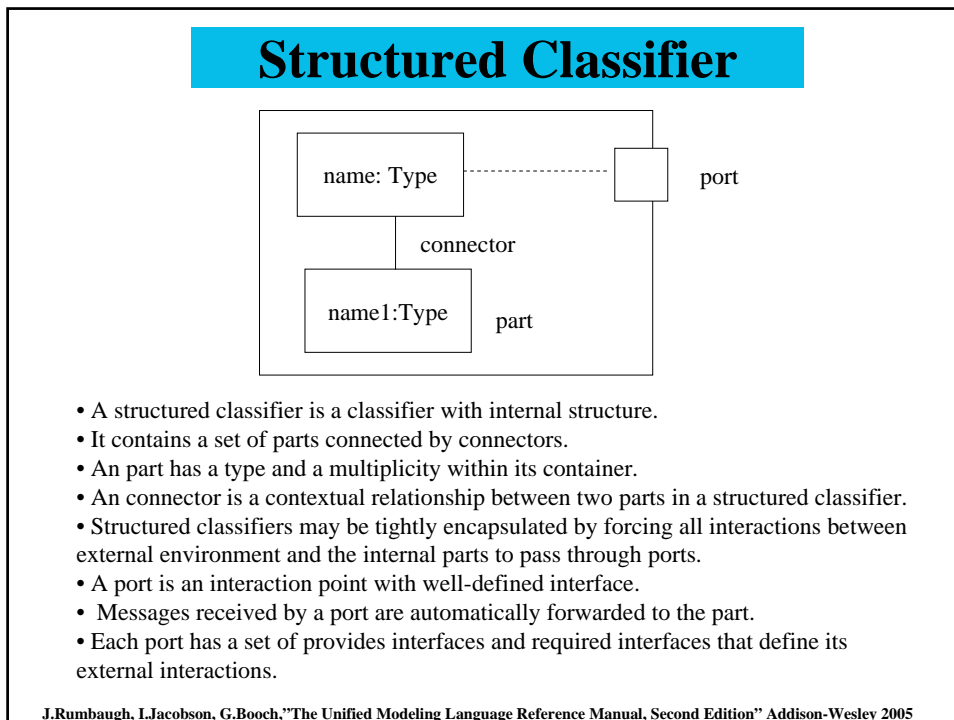
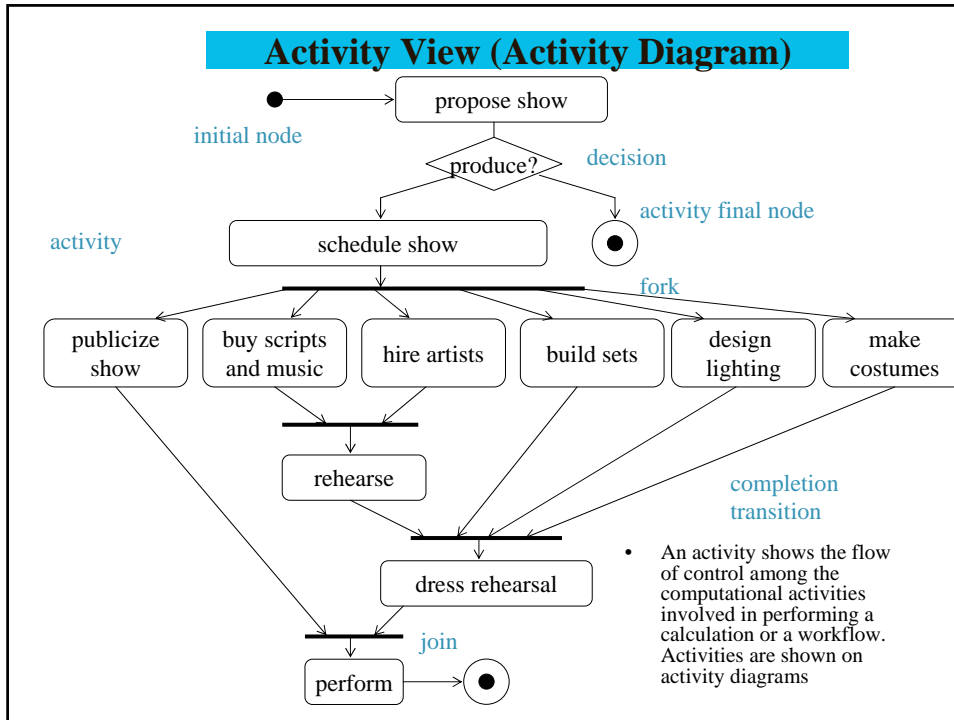
- **March 12**
  - 13:00 Unified Process and COMET
  - 14:30 Case Study of Elevator Control System  
(problem definition, use case model)
- **March 13**
  - 13:00 Case Study of Elevator Control System  
(finding analysis classes by developing a consolidated communication diagram)
  - 14:30 Case Study of Elevator Control System  
(sub-system structuring and task structuring)
- **March 14**
  - 13:00 Case Study of Elevator Control System  
( performance analysis)
  - 14:30 **UML2.0 and MDA**

## New Features of UML2.0

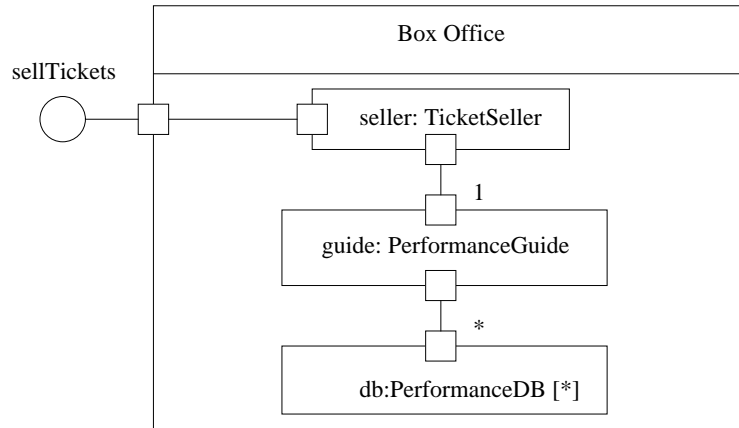
- Sequence Diagram constructs and notation based largely on the ITU ( International Telecommunication Union ) Message Sequence Chart standard, adapted to make it more object-oriented
- Decoupling of activity modeling concepts from state machines and use of notation popular in the business modeling community.
- Contextual modeling constructs for the internal composition of classes and collaborations. These constructs permit both loose and strict encapsulation and wiring of internal structures from smaller parts.
- Repositioning of components as design constructs and artifacts as physical entities that are deployed

### Structured Control constructs in a Sequence Diagram





## Design View (Internal structure diagram)

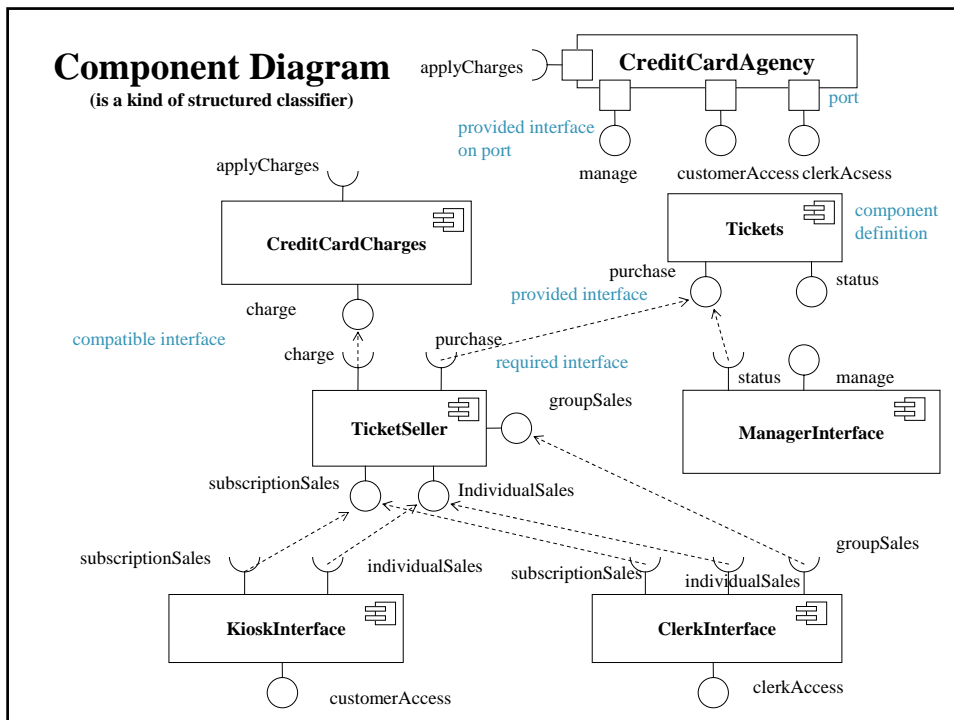
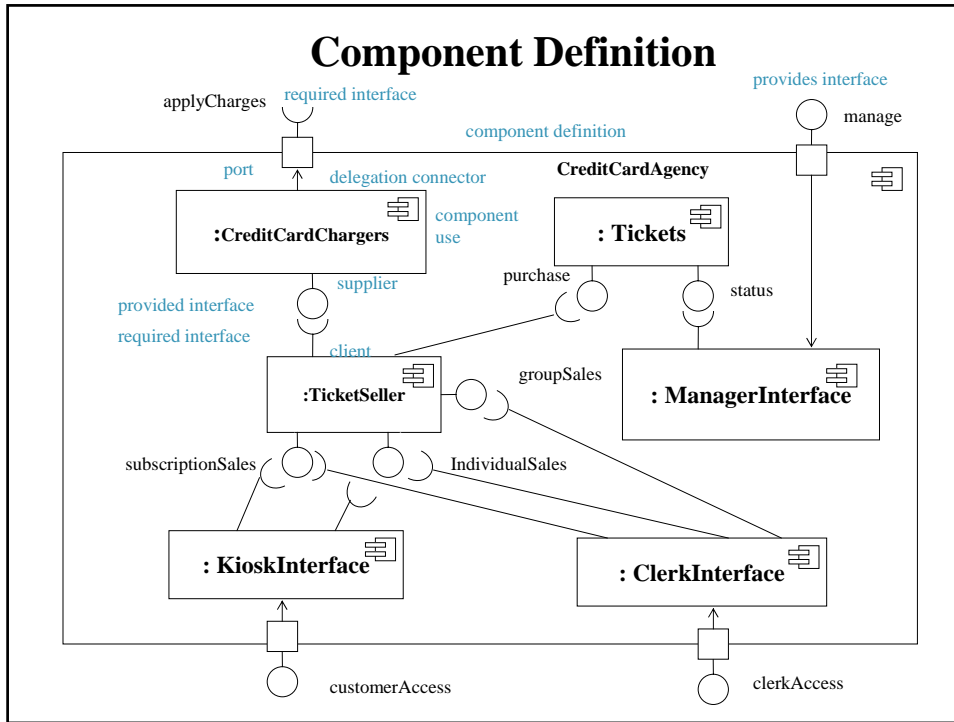


- Each port has a set of provides interfaces and required interfaces that define its external interactions. A provided interface specifies the services that a message to the port may request. A required interface specifies the services that a message from the port may require from the external environment.

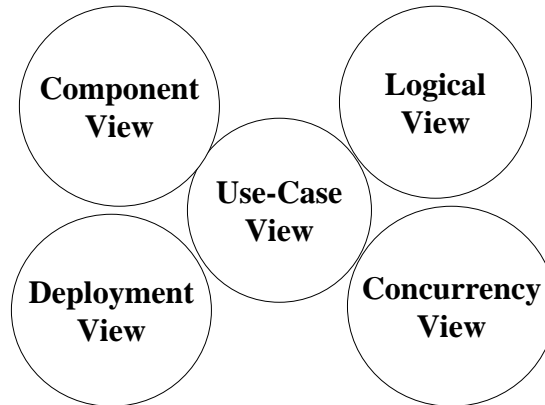
J.Rumbaugh, I.Jacobson, G.Booch, "The Unified Modeling Language Reference Manual, Second Edition" Addison-Wesley 2005

## Design View (component diagram)

- A component diagram is a kind of structured classifier, so its internal structure may be defined on an internal structure diagram.
- A component diagram shows the components in a system – that is, the software units from which the application is constructed. A small circle attached to a component or a class is a provided interface- a coherent set of services made available by a component or class.
- A small semicircle attached to a component or a class is a required interface – a statement that the component or class needs to obtain services from an element that provides them.



## Views of UML1.5



H.E. Eriksson and M. Penker, "UML Toolkit" John Wiley & Sons, Inc.

## UML2.0 Views

- Major Area, View
  - DiagramMain Concepts
- structural
  - static view : class diagram
  - design view : **internal structure (connector, interface, part, port, provided interface, role, required interface)**, collaboration diagram (connector, collaboration use, role), component diagram (component, dependency, port, provided interface, realization, required interface, subsystem)
  - use case view : usecase diagram
- dynamic
  - state machine view : state machine diagram
  - activity view : **activity diagram**
  - interaction view : **sequence diagram, communication diagram**
- physical
  - deployment view : deployment diagram
- model management
  - model management view : package diagram
  - profile : package diagram

## MDA and CBSD

- MDA ( Model Driven Architecture)
- CBSD (Component Based Software Development)

## MDA(1/2)

M3  
MetaMetaModel

MOF (Meta Object Facility)

M2  
MetaModel

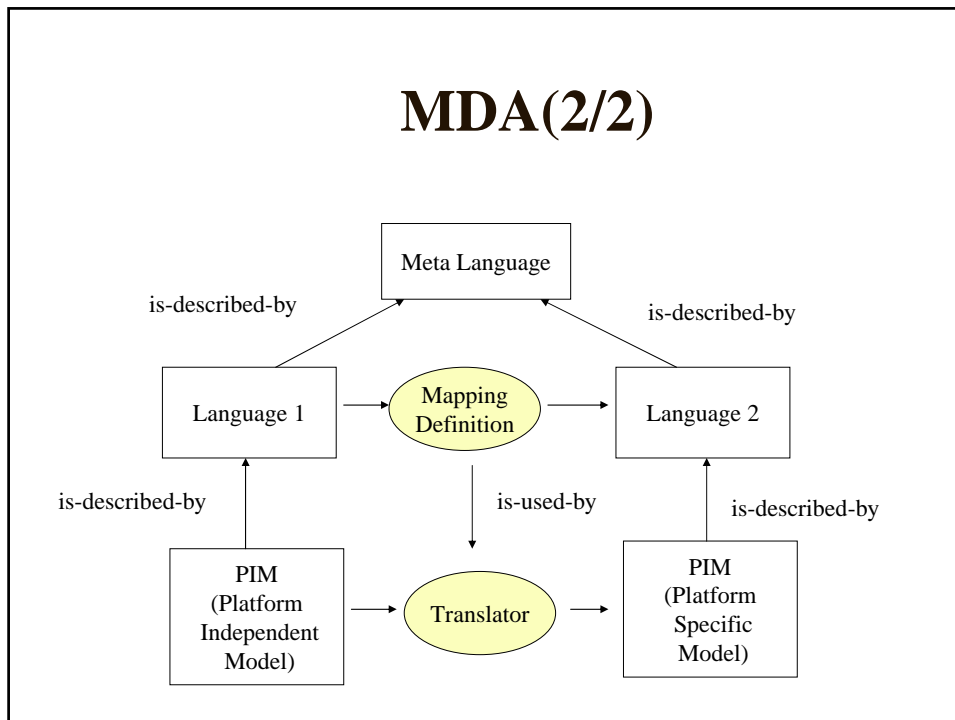
UML Class, UML Association, CWM Table

M1  
Model

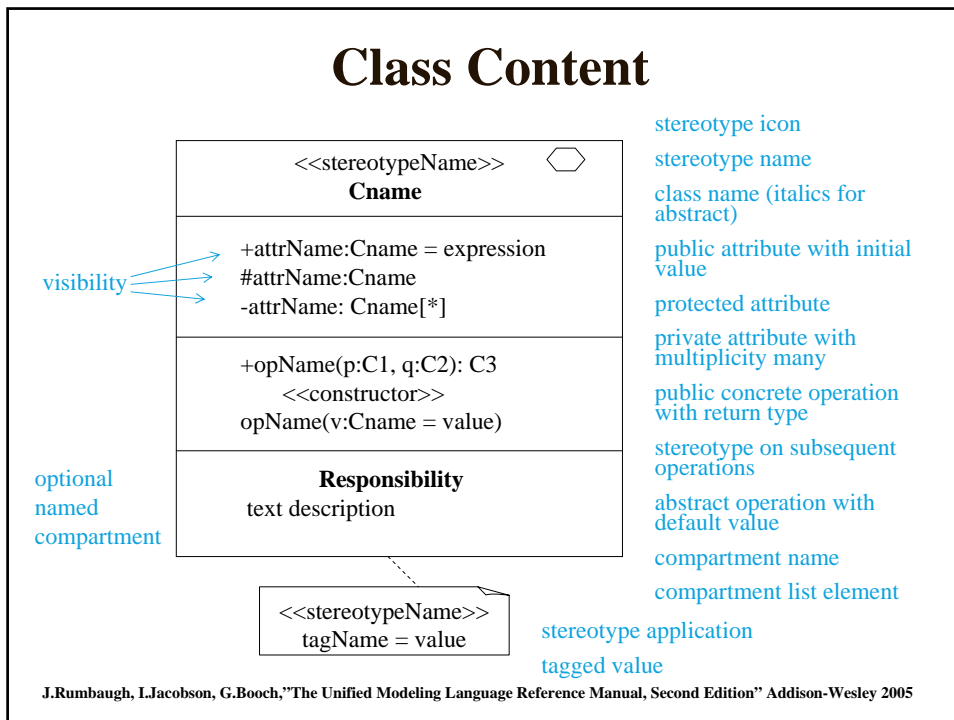
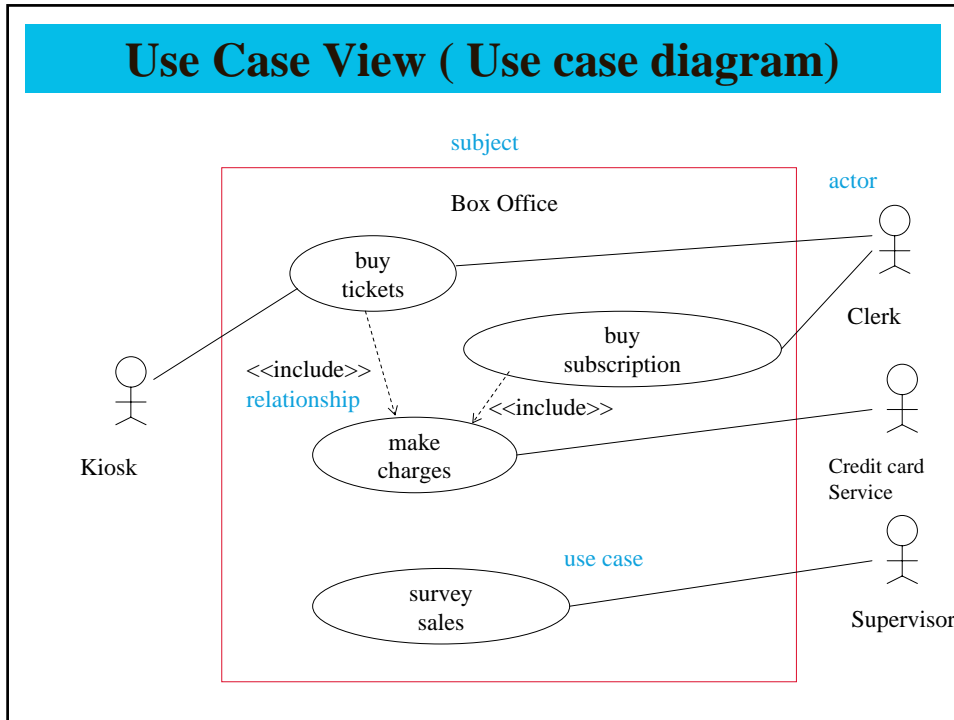
Class "Customer", Class "Account"

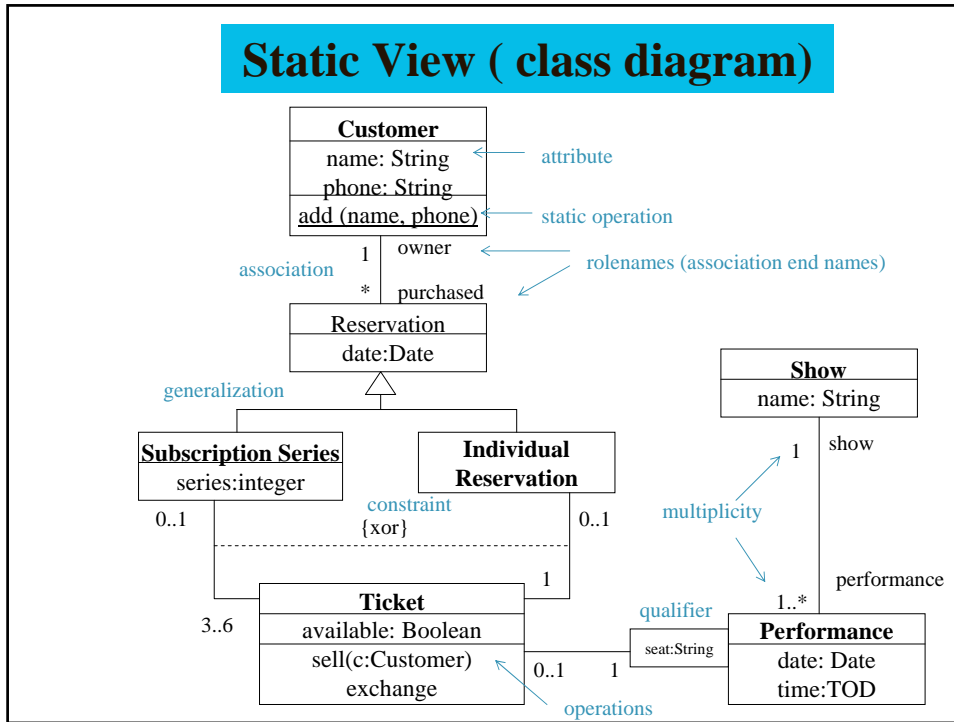
M0  
objects and data

Customer Jane Smith, Account 2989

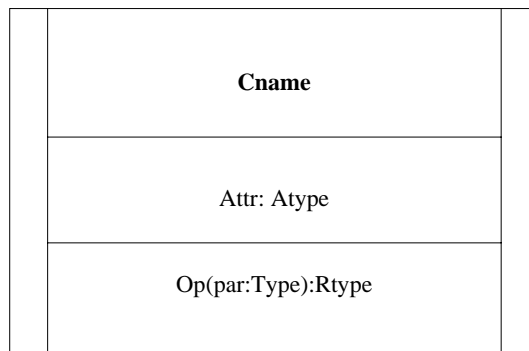


## Other Modifications



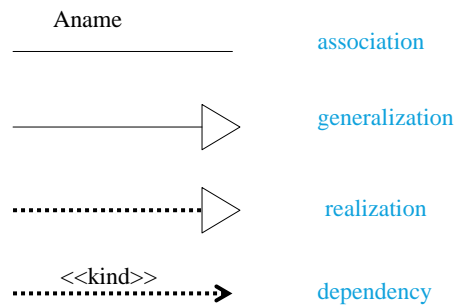


## Active Class



J.Rumbaugh, I.Jacobson, G.Booch, "The Unified Modeling Language Reference Manual, Second Edition" Addison-Wesley 2005

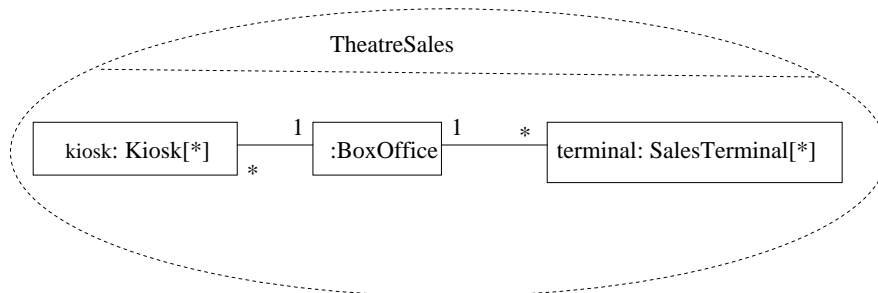
## Relationship



J.Rumbaugh, I.Jacobson, G.Booch, "The Unified Modeling Language Reference Manual, Second Edition" Addison-Wesley 2005

## Design View (collaboration diagram)

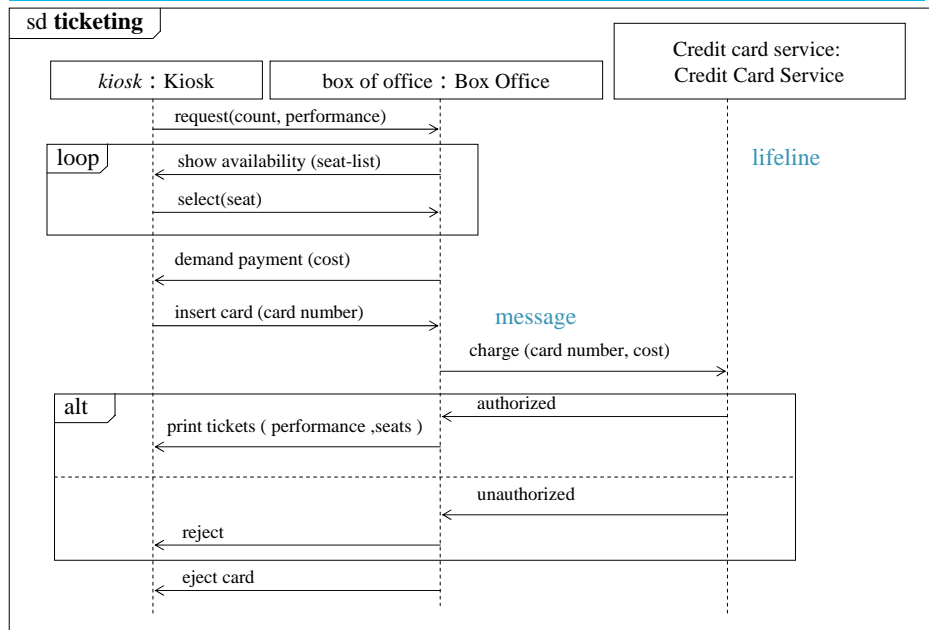
- A collaboration is a contextual relationship among a set of objects that work together to fulfill some purpose.
- It contains a collection of roles contextual slots within a generic pattern that can be played by, or bound to, individual objects.



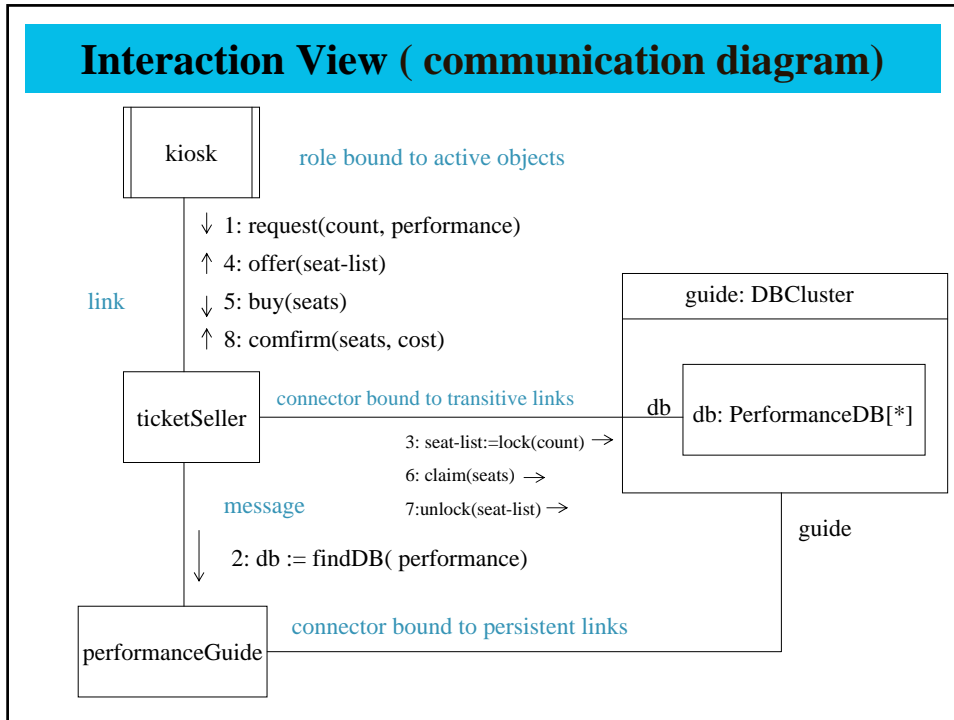
## Interaction View

- The interaction view describes sequence of message exchanges among the parts of a system.
- An interaction is based on a structured classifier or a collaboration.
- A role is a slot that may be filled by objects in a particular use of an interaction.
- Interaction view shows the flow of control across many objects and is displayed in two diagrams focused on different aspects: sequence diagrams and communication diagrams. The communication diagram is called a collaboration diagram in UML1.5.

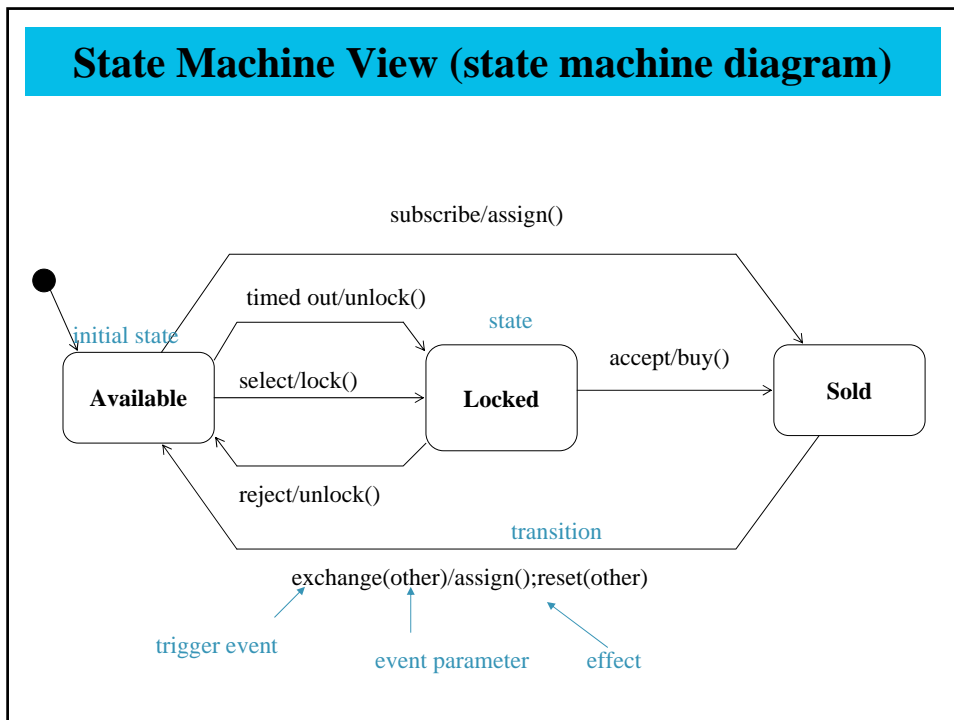
### Interaction View ( Sequence Diagram )

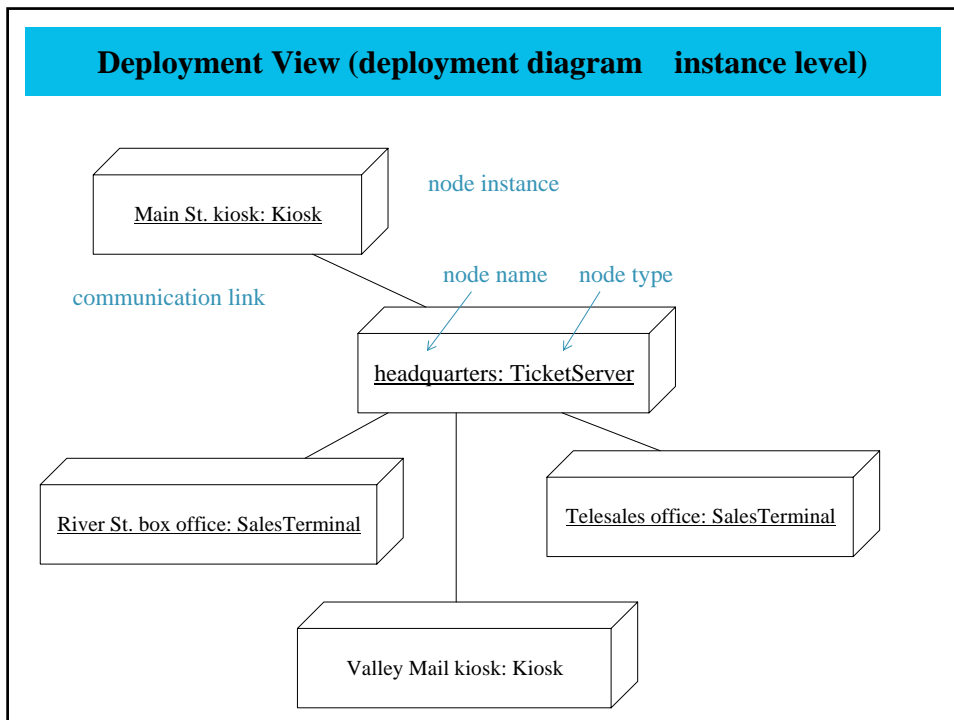
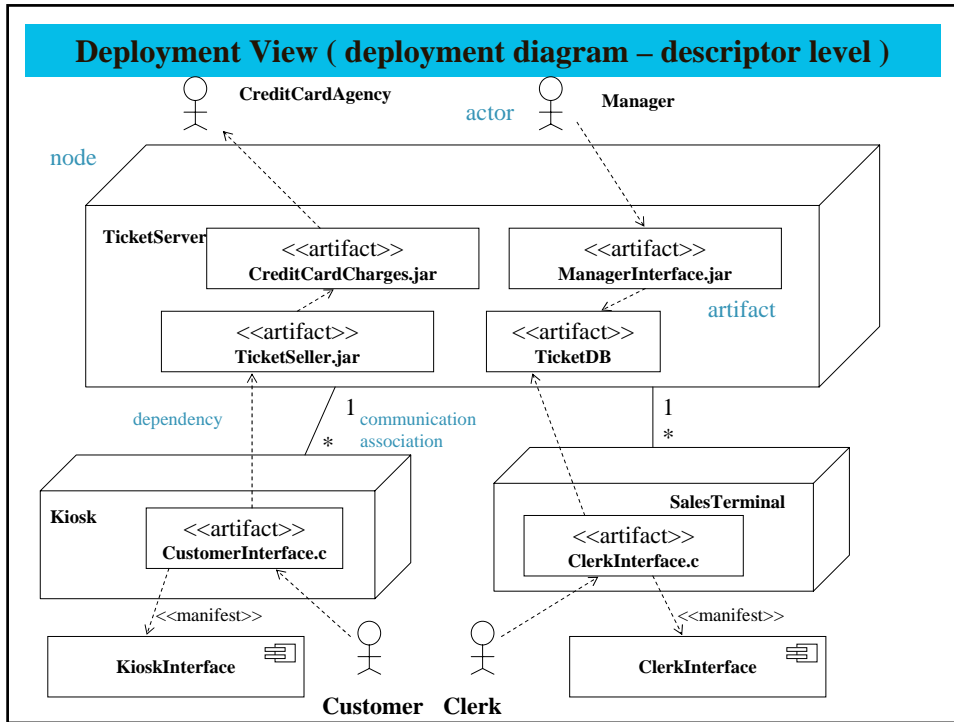


## Interaction View ( communication diagram)

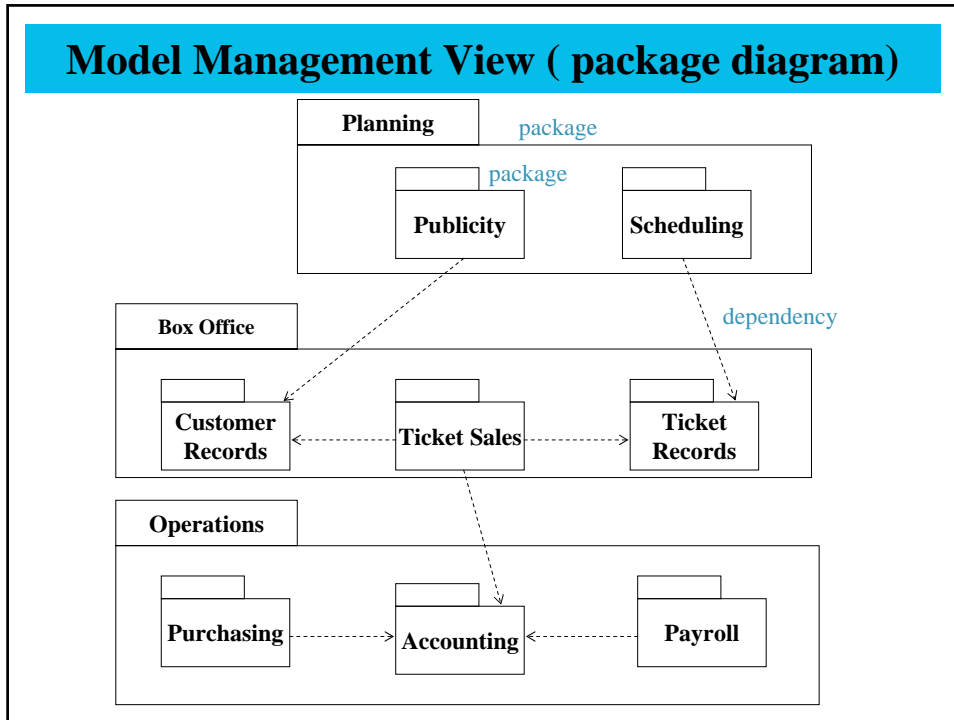


## State Machine View (state machine diagram)





## Model Management View ( package diagram)



## Model Management View ( Profile)

- The profile mechanism permits limited changes to UML without modifying the underlying metamodel.
- UML includes three main extensibility constructs: constraints, stereotypes, and tagged values

