UNIVERSITY OF INFORMATION TECHNOLOGY

Faculty of Information Systems

Chapter 4

Data and Process Modeling – Object Oriented Method

Dr. Cao Thi Nhan

LEARNING OBJECTIVES

1. Understand basic concepts of Activity Diagram, Class Diagram, Sequence Diagram, and State Diagram.
2. Have ability to create Activity Diagram, Class Diagram, Sequence Diagram, and State Diagram

CONTENT

1. Activity Diagram
2. Sequence Diagram
3. Class Diagram
4. State Machine

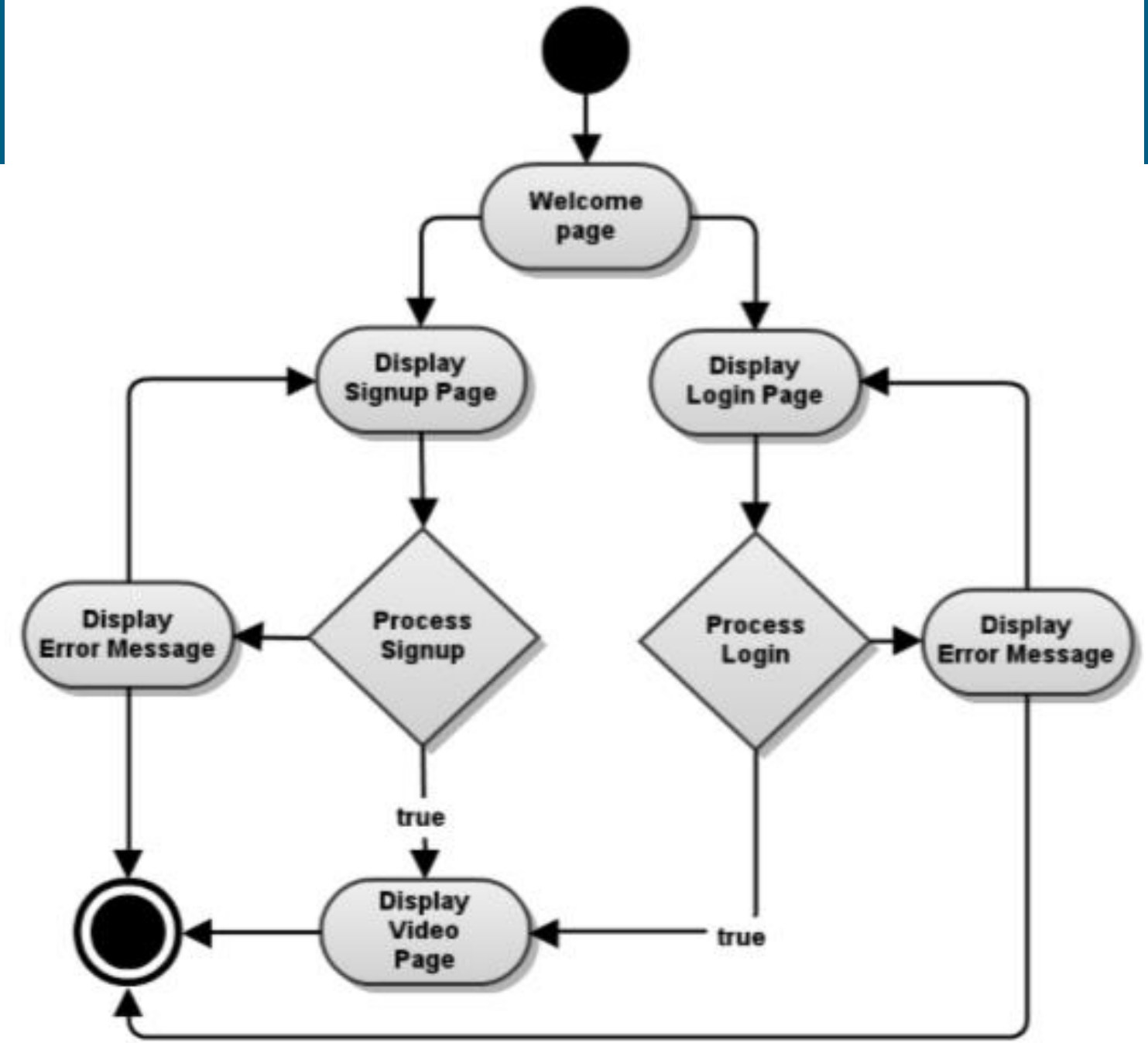
Activity Diagram

Activity Diagram

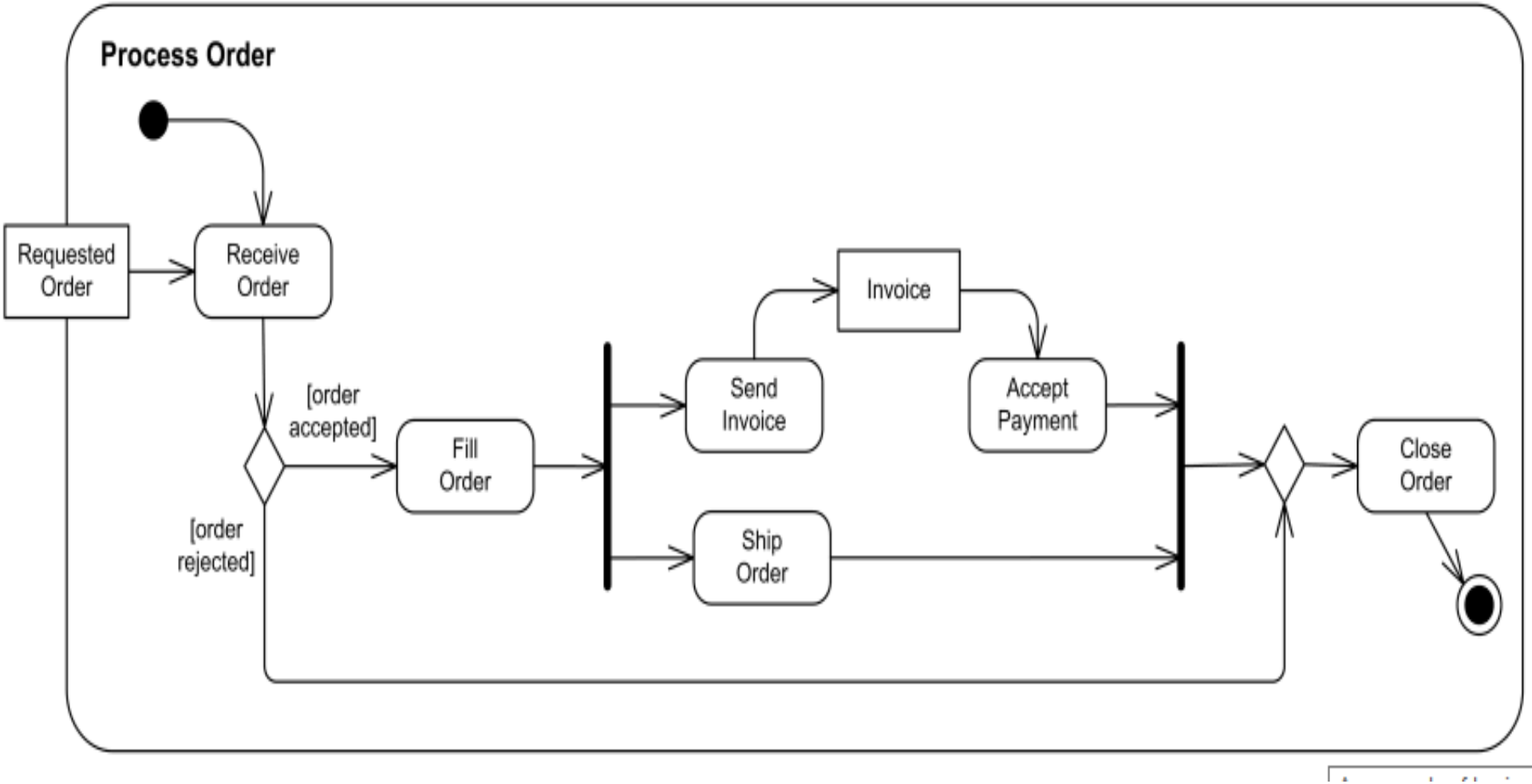
1. Introduction
2. Elements of an Activity Diagram
3. Guidelines for Creating Activity Diagrams

Activity diagram

- **Activity diagram** is UML behavior diagram which shows **flow of control** or **object flow** with emphasis on the sequence and conditions of the flow.
- Activity diagrams can be used to model everything from a high-level business workflow that involves many different use cases, to the details of an individual use case
- Activity diagrams describe the primary activities and the relationships among the activities in a process
- Activity diagram is used to describe use case based on Flow of events


Welcome page
Display Signup Page
Display Login Page
Display Error Message
Process Signup
Process Login
Display Error Message
true
Display Video Page
true

Activity diagram



Activity Diagram

1. Introduction
2. Elements of an Activity Diagram
3. Guidelines for Creating Activity Diagrams

Elements of an Activity Diagram

1. Action and activity
2. Object node
3. Control flow
4. Object flow
5. Initial node
6. Final-activity node
7. A final-flow node
8. A decision node
9. A merge node
10. A fork node
11. A join node
12. A swimlane

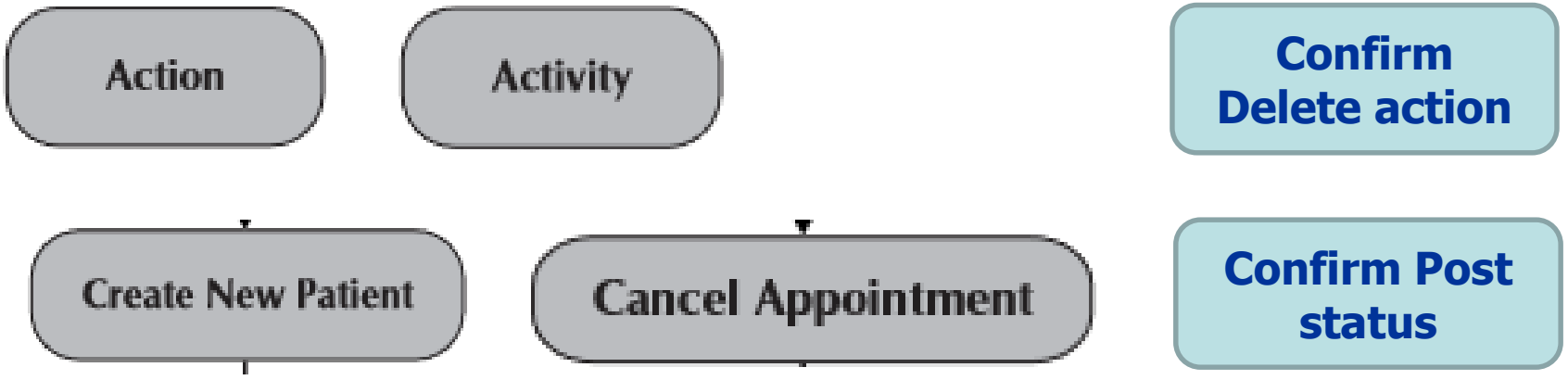
Elements of an Activity Diagram

1. Action and activity
2. Object node
3. Control flow
4. Object flow
5. Initial node
6. Final-activity node
7. A final-flow node
8. A decision node
9. A merge node
10. A fork node
11. A join node
12. A swimlane

Elements of an Activity Diagram

Action and activity:

- ✓ Action: Is a simple, nondecomposable piece of behavior.
- ✓ Activity: a set of actions
- ✓ Is labeled by its name (Verb – Noun)
- ✓ Actions and activities can represent manual or computerized behavior



Elements of an Activity Diagram

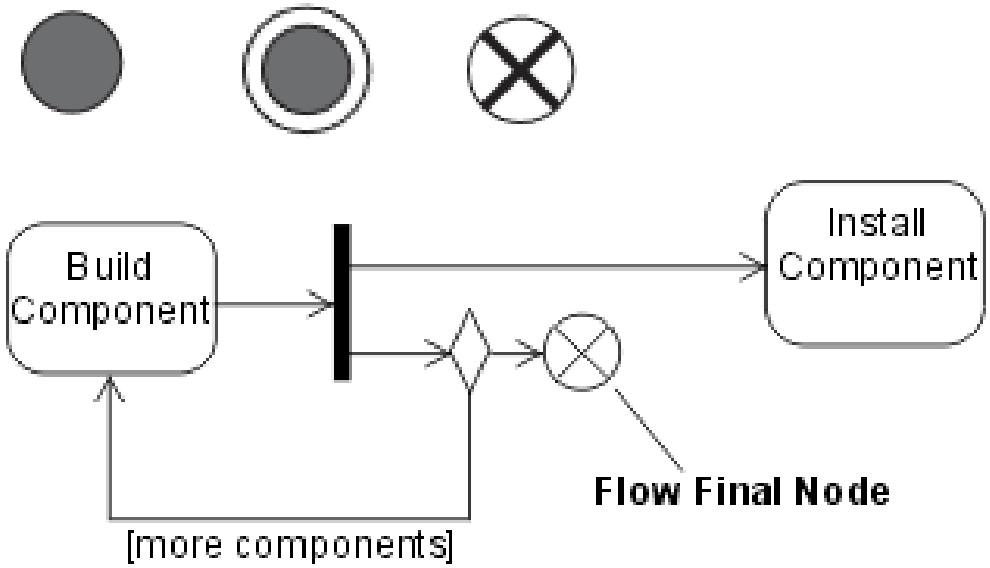
1. Action and activity
2. Object node
3. Control flow
4. Object flow
5. Initial node
6. Final-activity node
7. A final-flow node
8. A decision node
9. A merge node
10. A fork node
11. A join node
12. A swimlane

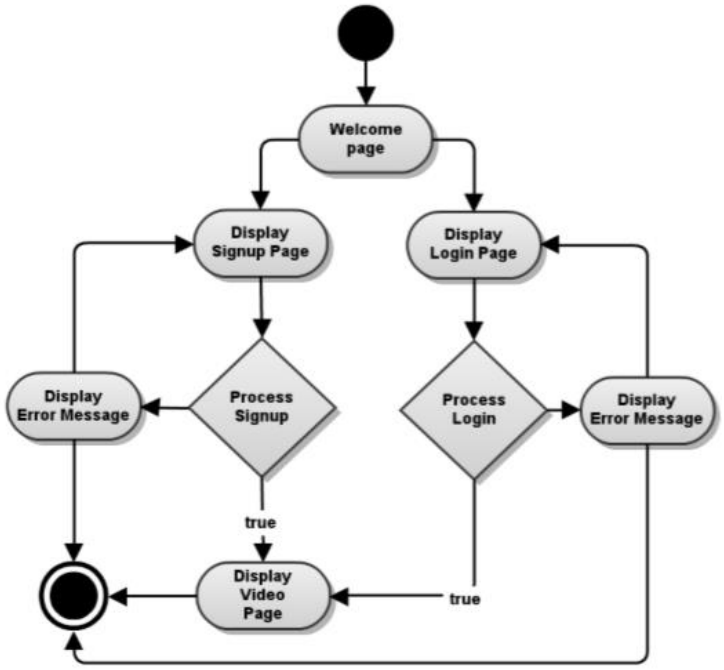
Elements of an Activity Diagram

Initial node: Portrays the beginning of a set of actions or activities

Final-activity node: Is used to stop all control flows and object flows in an activity (or action).

A final-flow node: Is used to stop a specific control flow or object flow





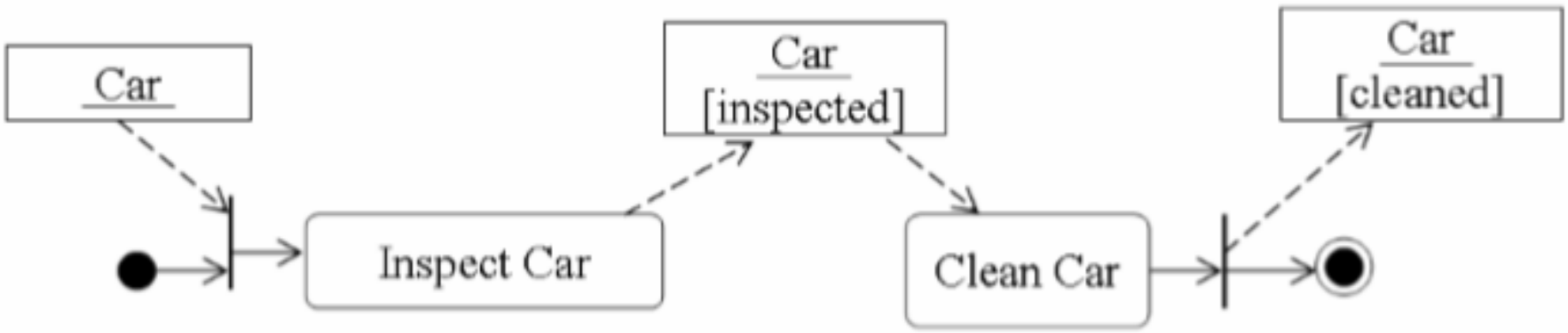
Elements of an Activity Diagram

1. Action and activity
2. Object node
3. Control flow
4. Object flow
5. Initial node
6. Final-activity node
7. A final-flow node
8. A decision node
9. A merge node
10. A fork node
11. A join node
12. A swimlane

Elements of an Activity Diagram

Object node:

- ✓ Is used to represent an object that is connected to a set of object flows.
- ✓ Is labeled by its class name
- ✓ Business case inspect and clean car [Jens Brüning, Peter Forbrig, *Behaviour of flow operators connected with object flows in workflow specifications*]



Elements of an Activity Diagram

1. Action and activity
2. Object node
3. Control flow
4. Object flow
5. Initial node
6. Final-activity node
7. A final-flow node
8. A decision node
9. A merge node
10. A fork node
11. A join node
12. A swimlane

Elements of an Activity Diagram

Control flow:

- ✓ Shows the sequence of execution 

Object flow:

- ✓ Shows the flow of an object from one activity to another active 

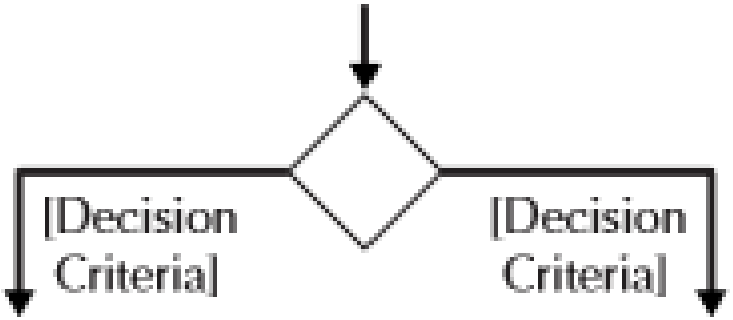
Elements of an Activity Diagram

1. Action and activity
2. Object node
3. Control flow
4. Object flow
5. Initial node
6. Final-activity node
7. A final-flow node
8. A decision node
9. A merge node
10. A fork node
11. A join node
12. A swimlane

Elements of an Activity Diagram

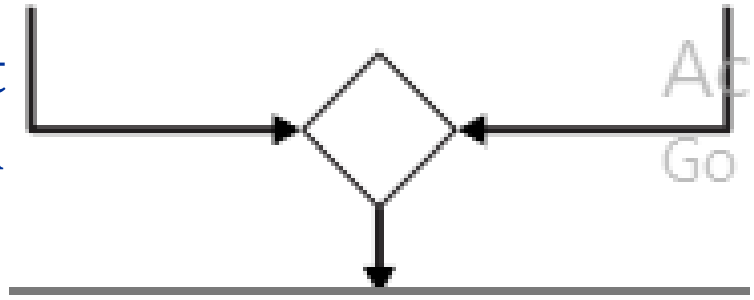
A decision node:

- ✓ Is used to represent a test condition
- ✓ Is labeled with the decision criteria



A merge node:

- ✓ Is used to bring back together different decision paths that were created using decision node

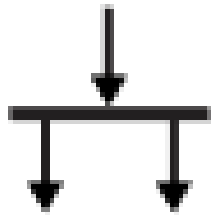


Elements of an Activity Diagram

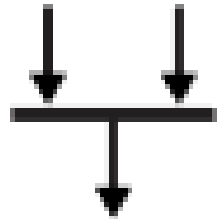
1. Action and activity
2. Object node
3. Control flow
4. Object flow
5. Initial node
6. Final-activity node
7. A final-flow node
8. A decision node
9. A merge node
10. A fork node
11. A join node
12. A swimlane

Elements of an Activity Diagram

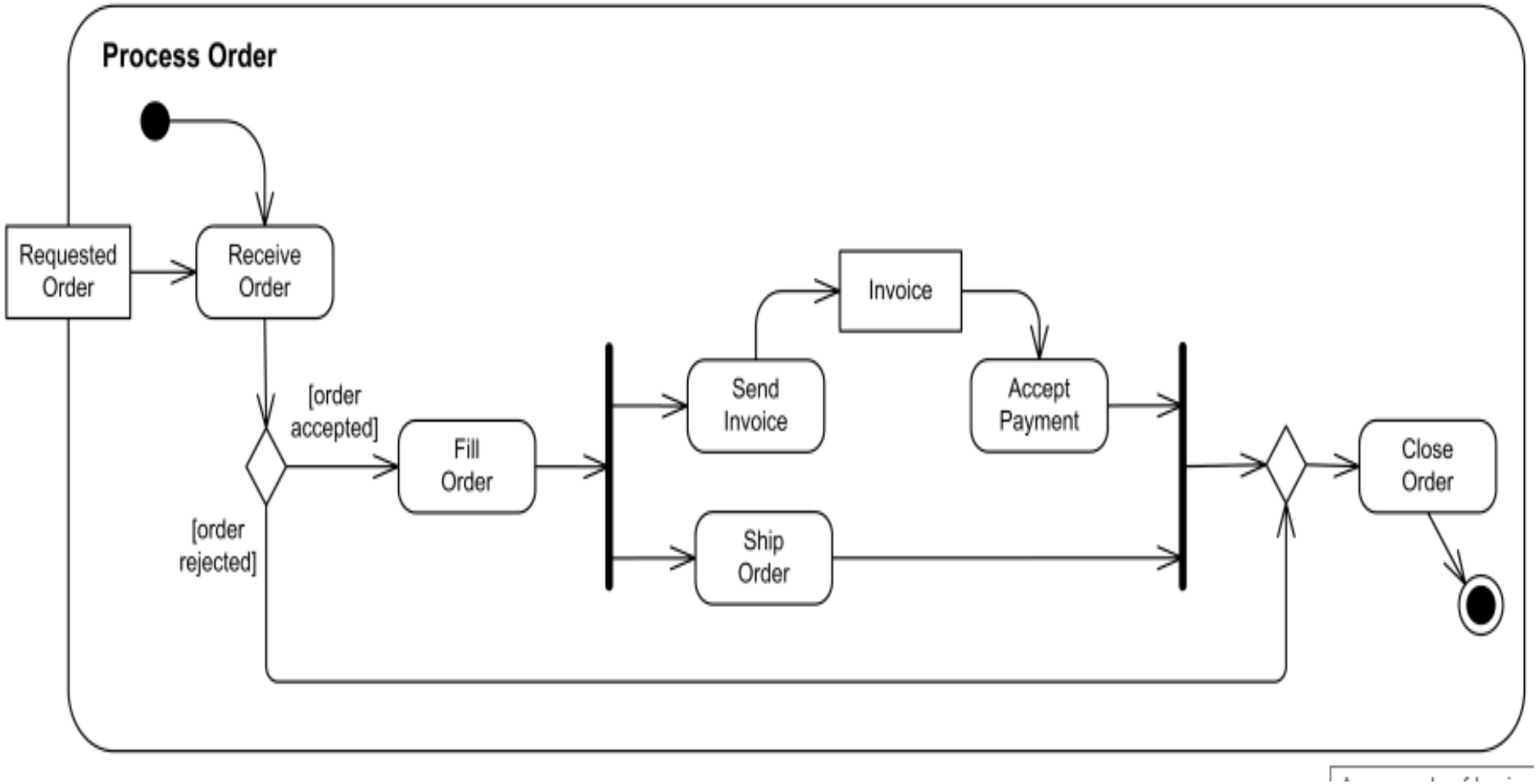
A fork node: Is used to split behavior into a set of parallel or concurrent flows of activities



A join node: Is used to bring back together a set of parallel or concurrent flows of activities



Elements of an Activity Diagram

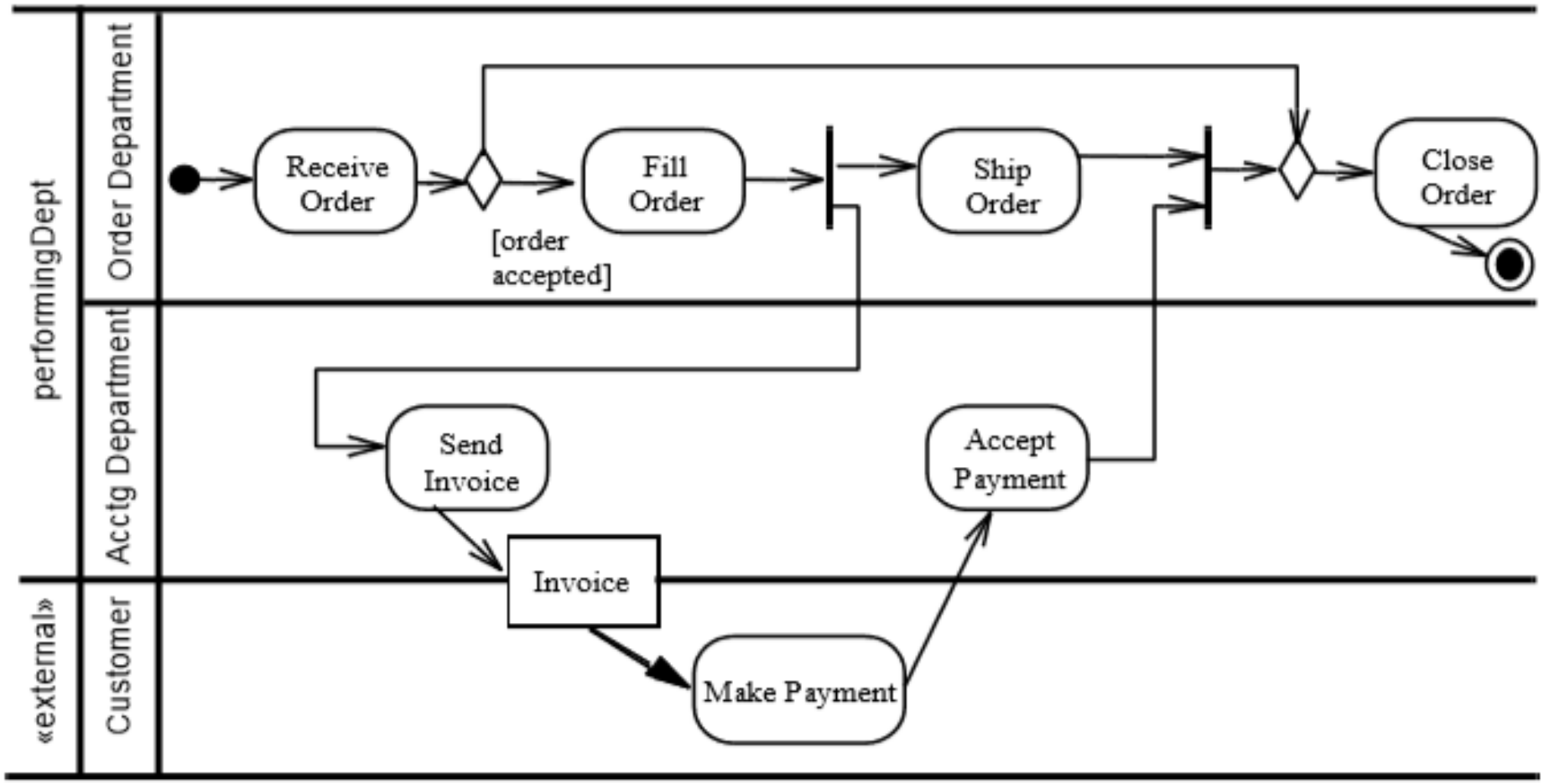


Elements of an Activity Diagram

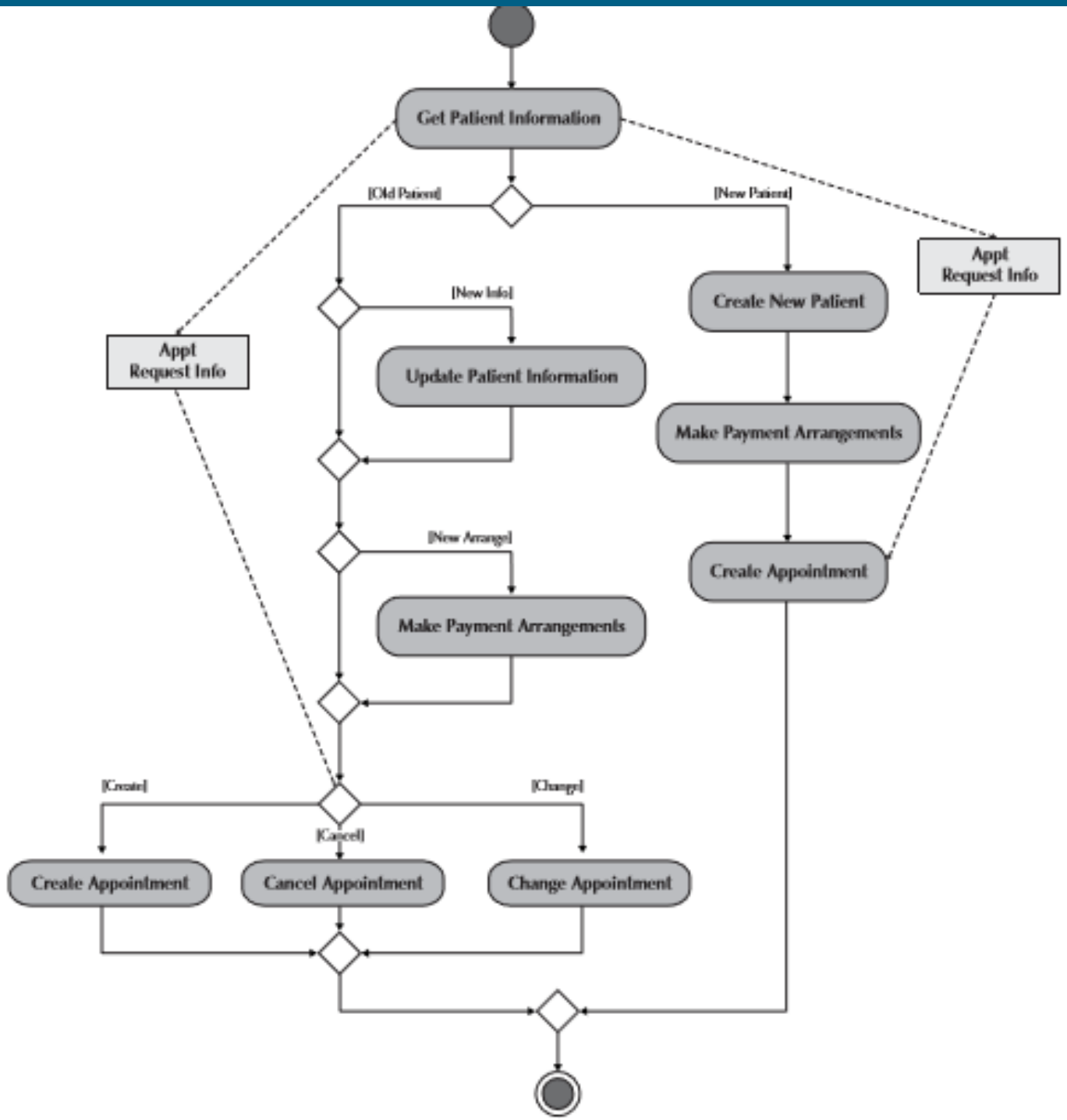
A swimlane:

- ✓ Is used to break up an activity diagram into rows and columns to assign the individual activities to the individuals or objects that are responsible for executing the activity
- ✓ Is labeled with the name of the individual or object responsible

Elements of an Activity Diagram



Activity Diagram for the ***Manage Appointment*** use case




QUESTIONS
ANSWERS

CONTENT

1. Activity Diagram
2. Sequence Diagram
3. Class Diagram
4. State Machine

Sequence Diagram

Sequence Diagrams

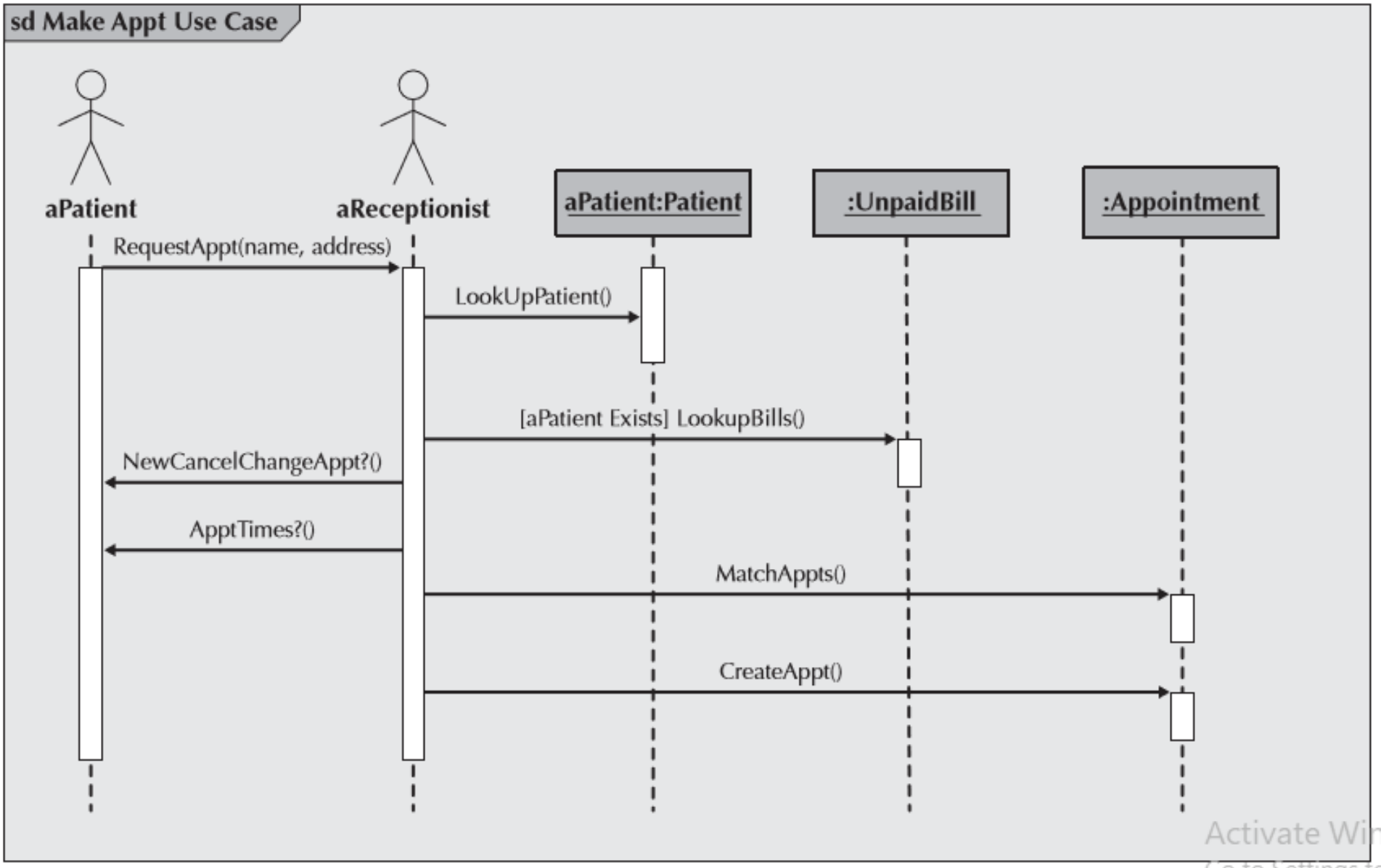
1. Introduction
2. Elements of a Sequence Diagram
3. Guidelines for Creating Sequence Diagrams

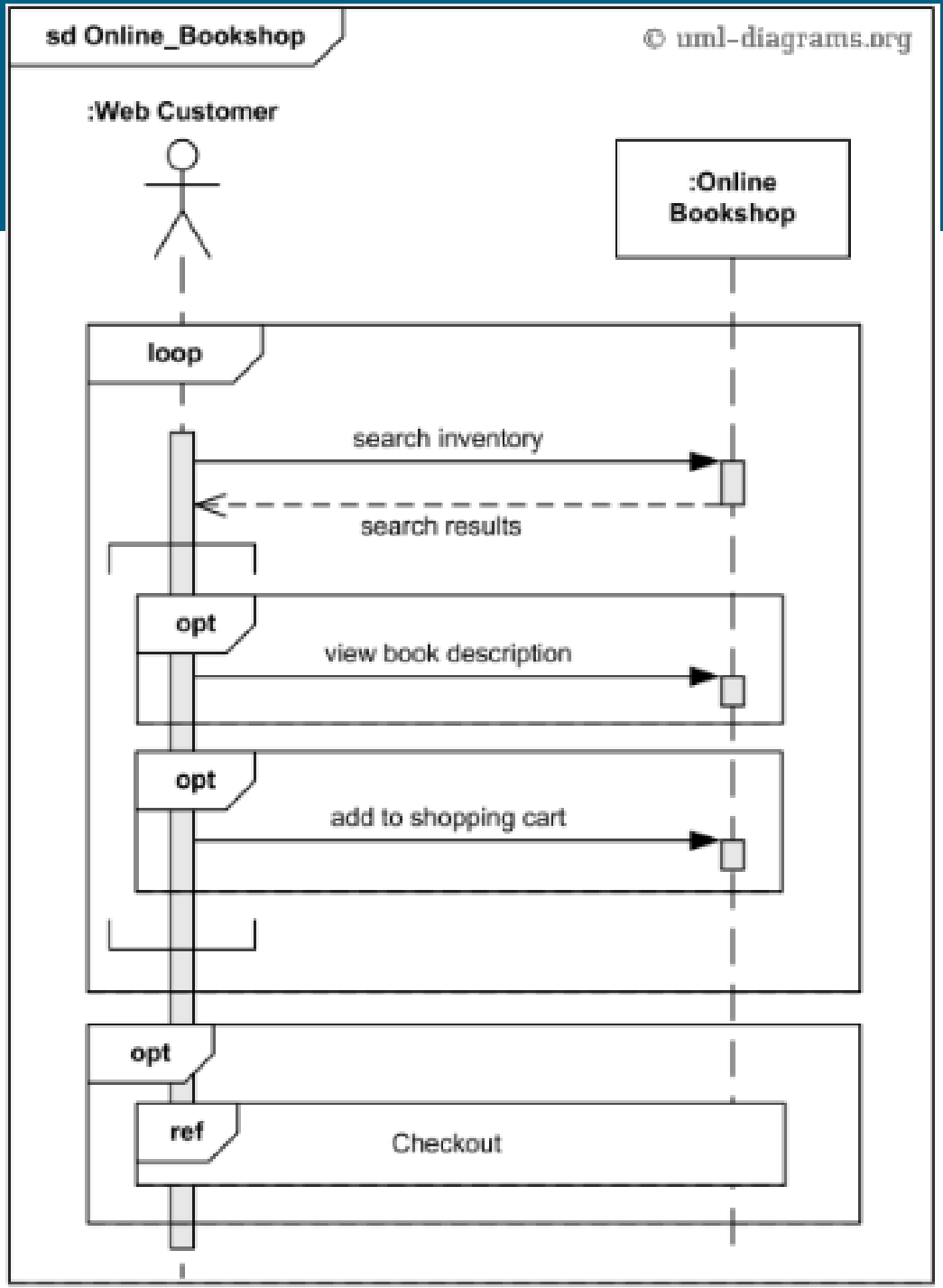
Sequence Diagrams

Introduction

- ✓ Sequence diagram shows the objects that participate in a use case and the messages that pass between them over time for one use case
- ✓ It can be a *generic sequence diagram* that shows all possible scenarios for a use case
- ✓ Usually analyst develops a set of instance sequence diagrams, each of which depicts a single scenario within the use case.
- ✓ The design diagrams are very implementation specific, often including database objects or specific user interface components as the objects.

Sequence Diagrams




sd Online_Bookshop
© uml-diagrams.org
:Web Customer
:Online Bookshop
loop
search inventory
search results
opt
view book description
opt
add to shopping cart
opt
ref
Checkout

Sequence Diagrams

1. Introduction
2. Elements of a Sequence Diagram
3. Guidelines for Creating Sequence Diagrams

Sequence Diagrams

Elements of a Sequence Diagram

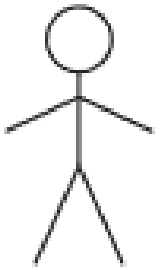
1. An actor
2. An object
3. A Lifeline
4. An execution occurrence
5. A Message
6. A guard condition
7. For object destruction
8. A frame

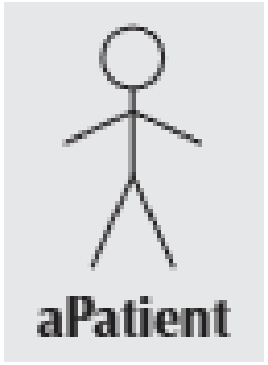
Sequence Diagrams

Elements of a Sequence Diagram

1. An actor:

- ✓ Is a person or system that derives benefit from and is external to the system.
- ✓ Participates in a sequence by sending and/or receiving messages





Sequence Diagrams

Elements of a Sequence Diagram

2. An object:

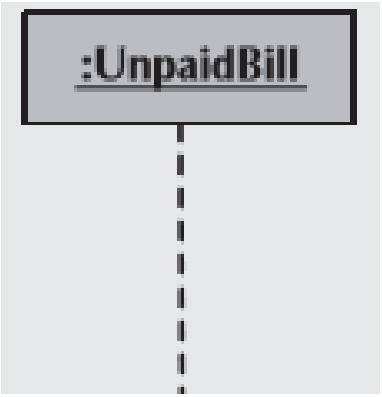
- ✓ Participates in a sequence by sending and/or receiving messages

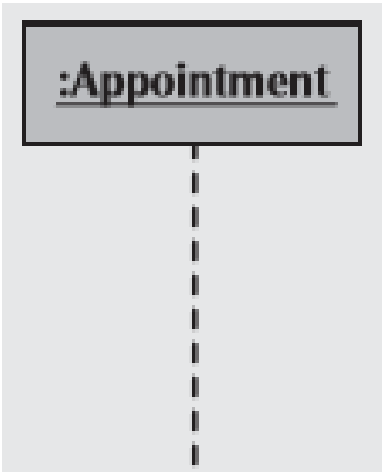
Sequence Diagrams

Elements of a Sequence Diagram

3. A lifeline:

- ✓ The life of an object during a sequence
- ✓ Contains an X at the point at which the class no longer interacts



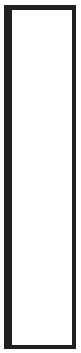


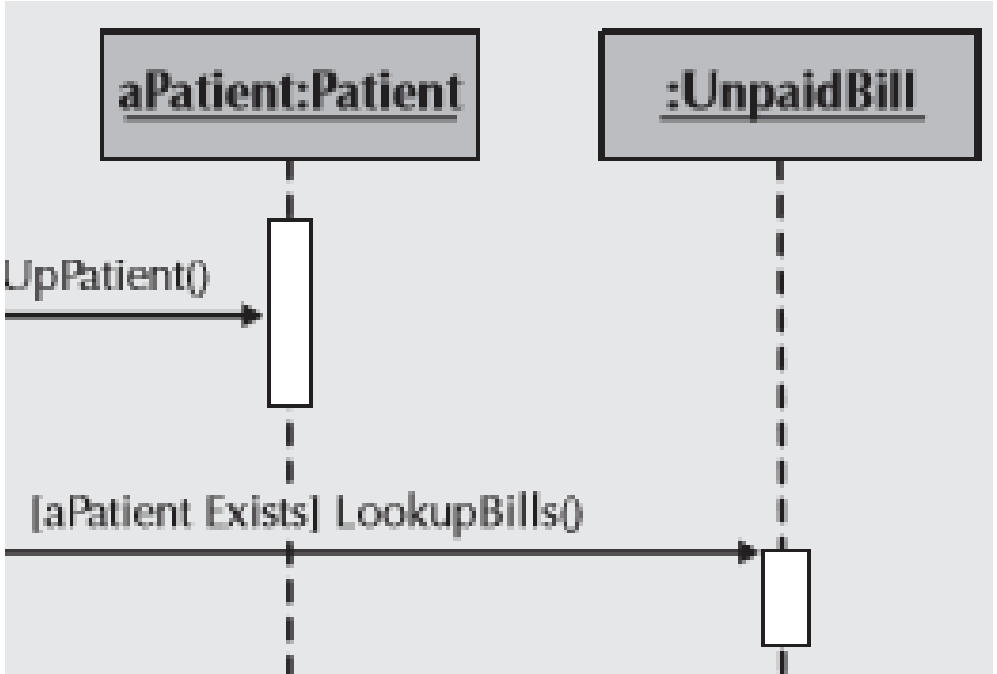
Sequence Diagrams

Elements of a Sequence Diagram

4. An execution occurrence:

- ✓ Is a long narrow rectangle placed atop a lifeline
- ✓ Denotes when an object is sending or receiving messages



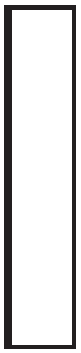


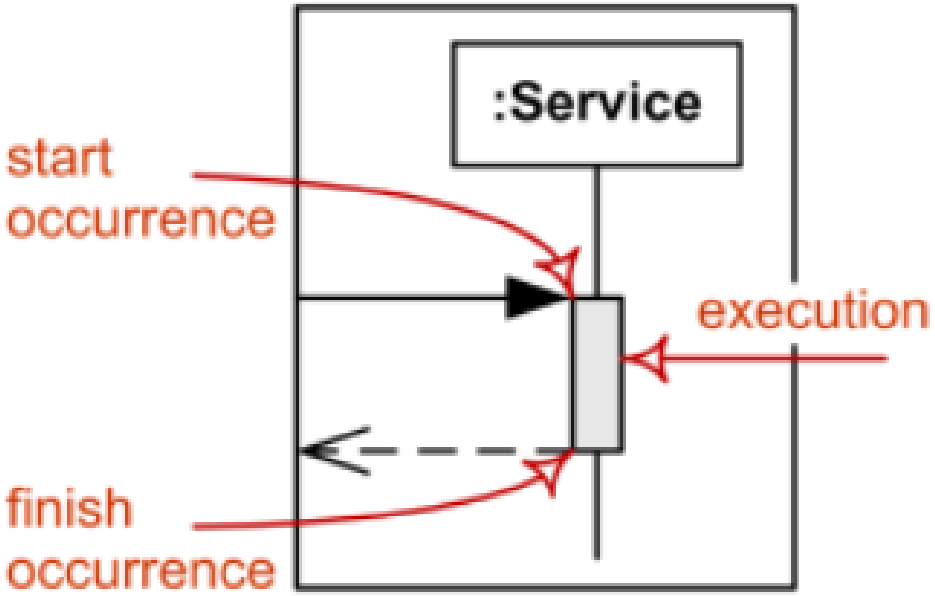
Sequence Diagrams

Elements of a Sequence Diagram

4. An execution occurrence:

- ✓ Is a long narrow rectangle placed atop a lifeline
- ✓ Denotes when an object is sending or receiving messages



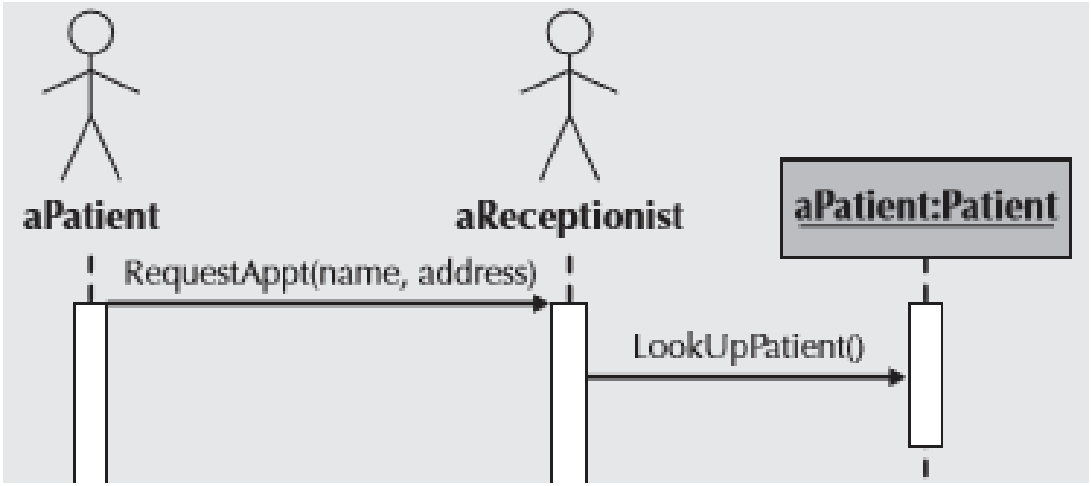


Sequence Diagrams

Elements of a Sequence Diagram

5. A message:

- ✓ Conveys information from one object to another one
- ✓ An operation call is labeled with the message being sent and *a solid arrow*, whereas a return is labeled with the value being returned and shown as *a dashed arrow*

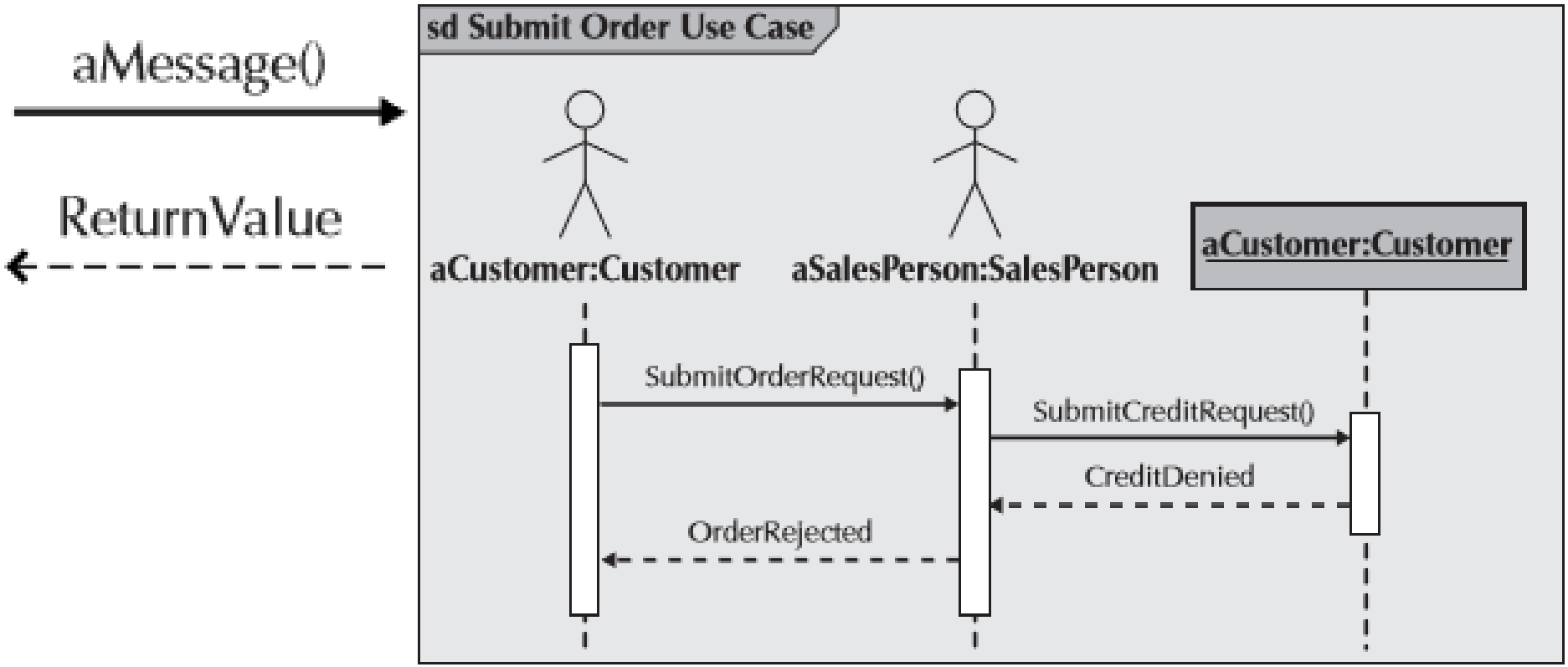


Sequence Diagrams

Elements of a Sequence Diagram

5. A message:

✓ Conveys information from one object to another one

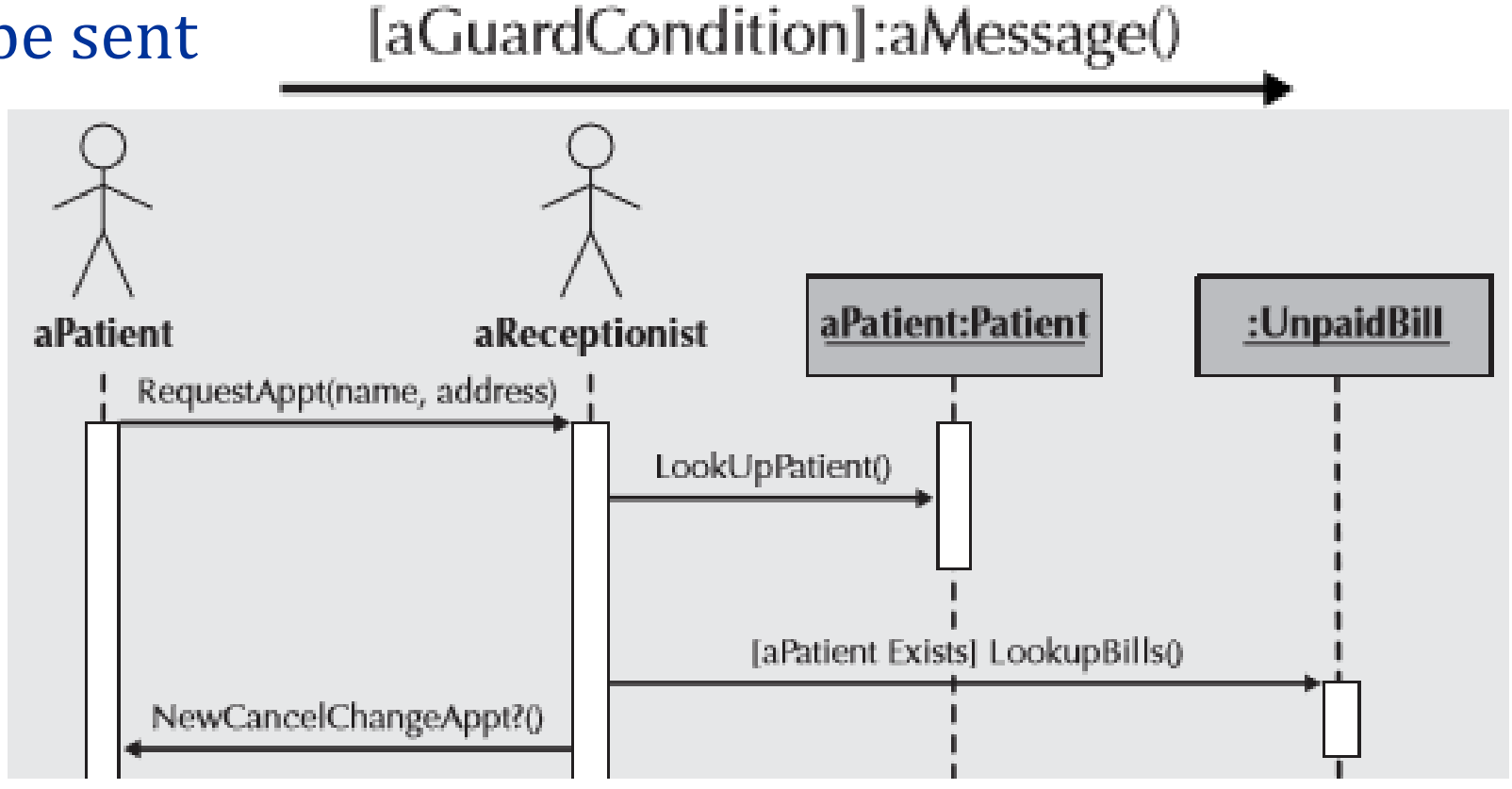


Sequence Diagrams

Elements of a Sequence Diagram

6. A guard condition:

- ✓ Represents a test that must be met for the message to be sent

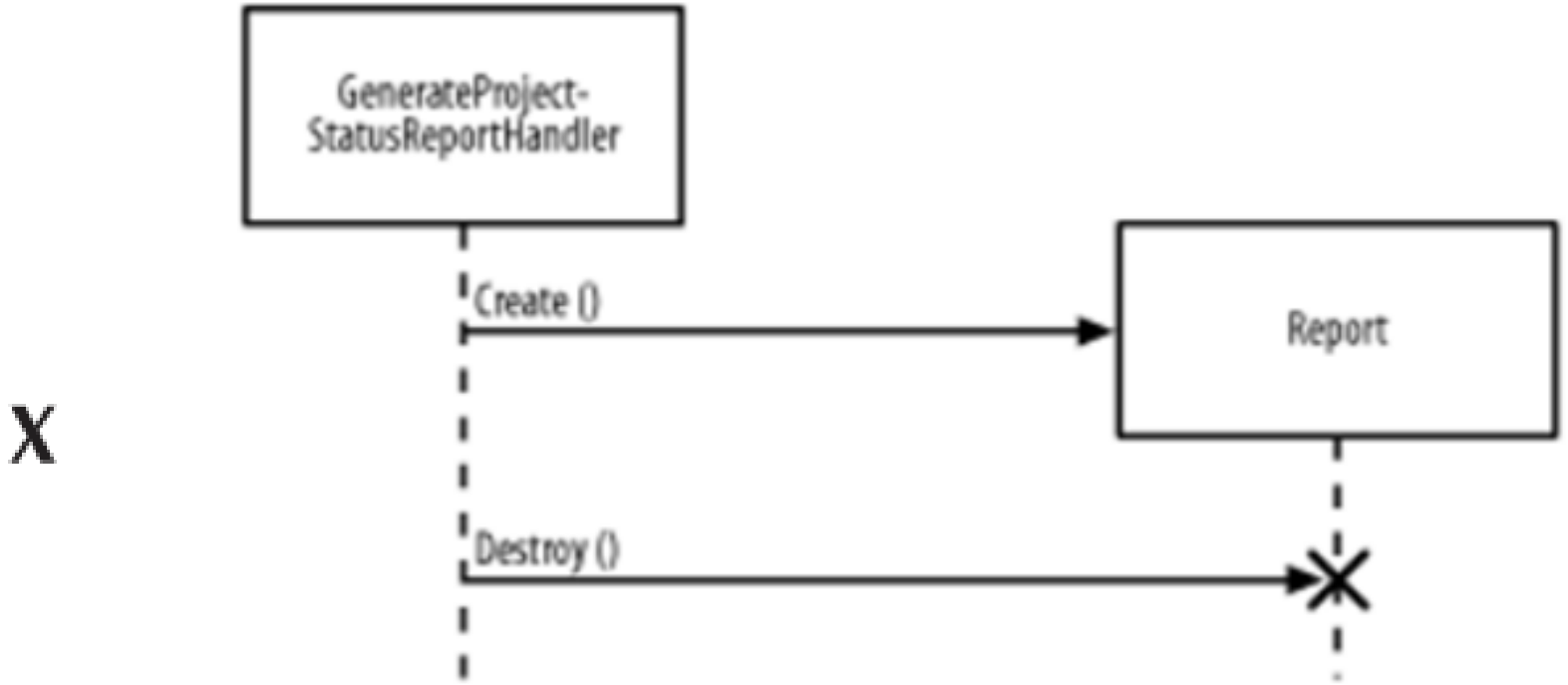


Sequence Diagrams

Elements of a Sequence Diagram

7. For object destruction:

- ✓ An X is placed at the end of an object's lifeline to show that it is going out of existence

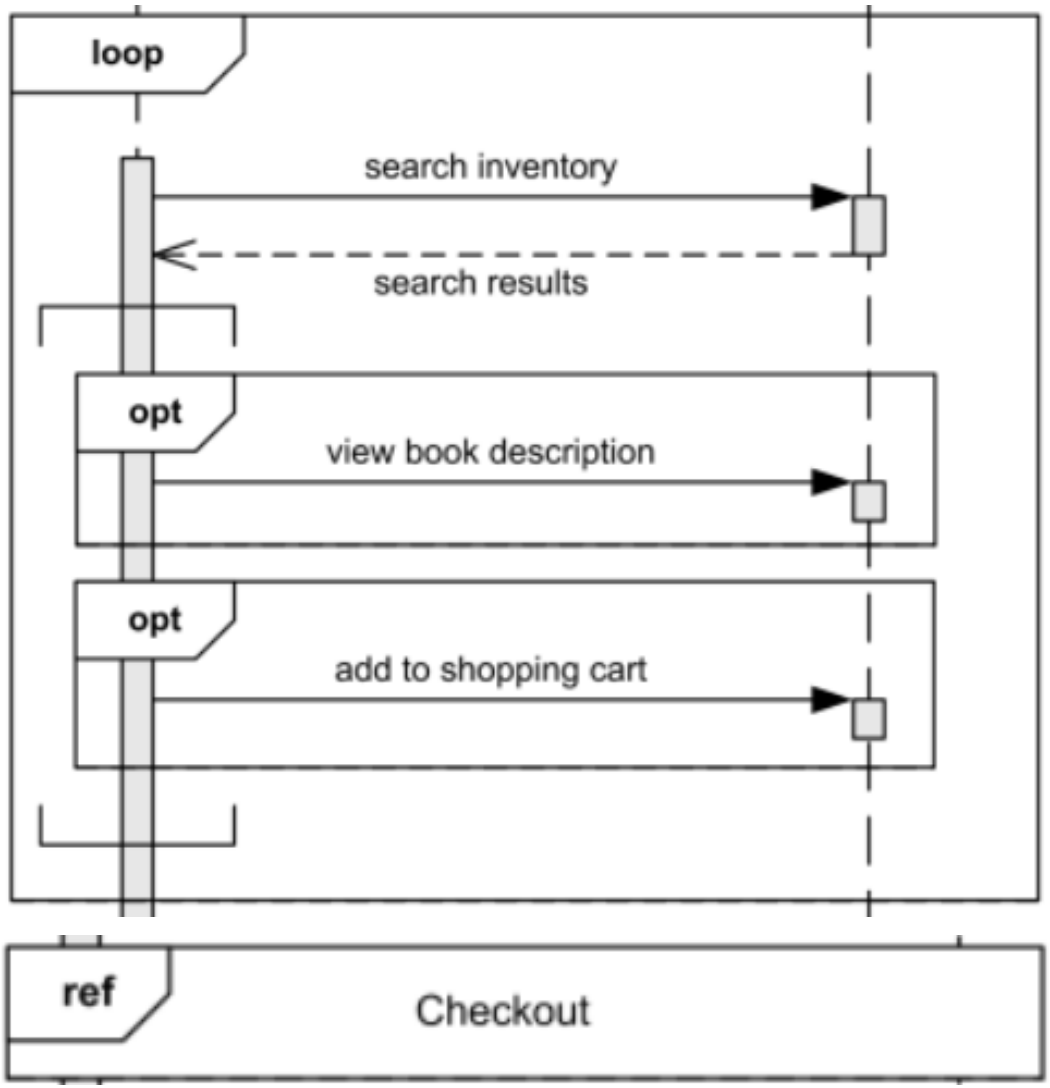


Sequence Diagrams

Elements of a Sequence Diagram

8. A frame:

- ✓ Indicates the context of the sequence diagram



Sequence Diagrams

Elements of a Sequence Diagram

1. An actor
2. An object
3. A Lifeline
4. An execution occurrence
5. A Message
6. A guard condition
7. For object destruction
8. A frame

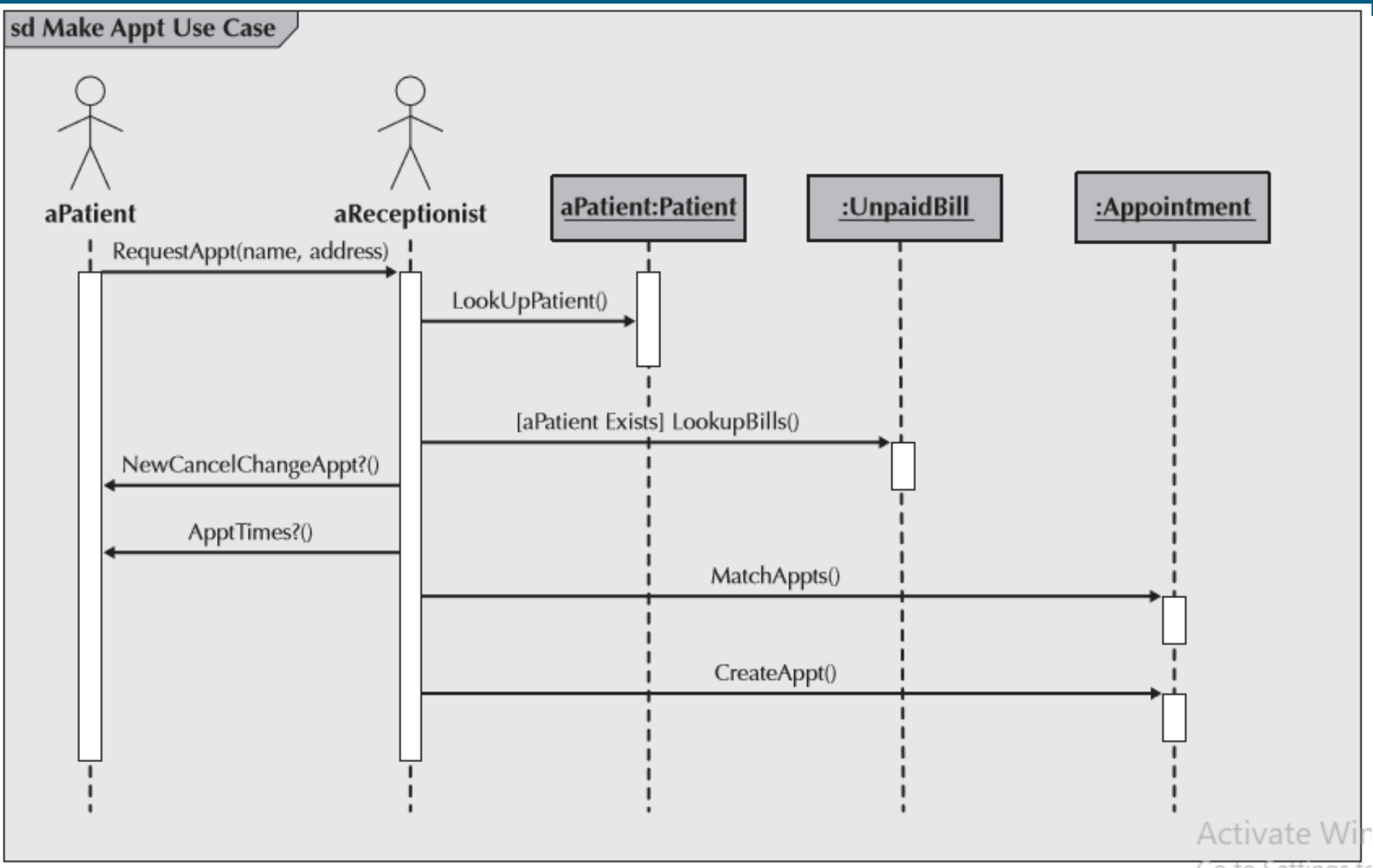
Sequence Diagrams

1. Introduction
2. Elements of a Sequence Diagram
3. Guidelines for Creating Sequence Diagrams

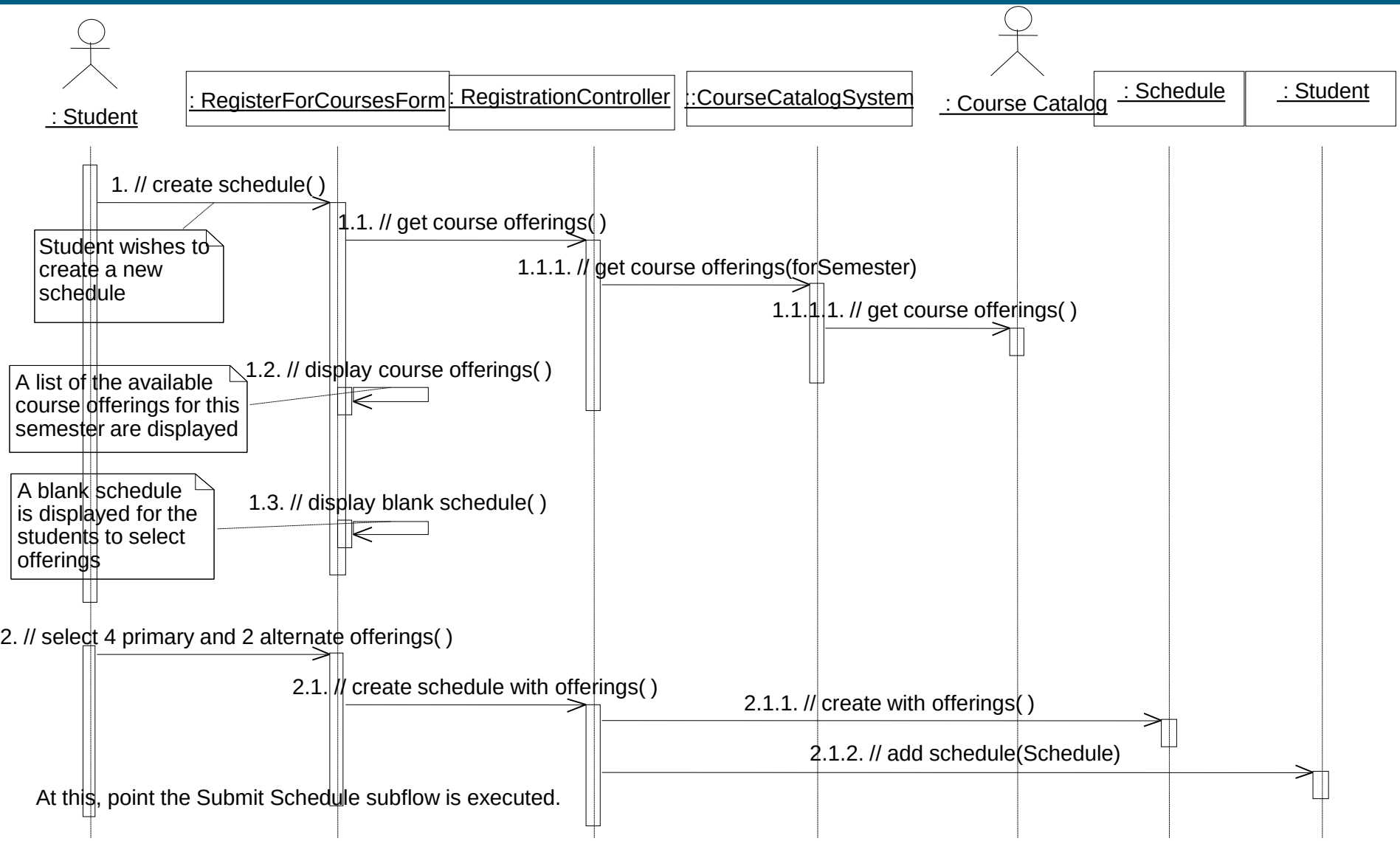
Guidelines for Creating Sequence Diagrams

1. Set context: use-case scenario
2. Identify Actors and Objects: the actors and objects that interact with each other during the use-case scenario
3. Set Lifeline
4. Add Messages
5. Place Execution Occurrence
6. Validate

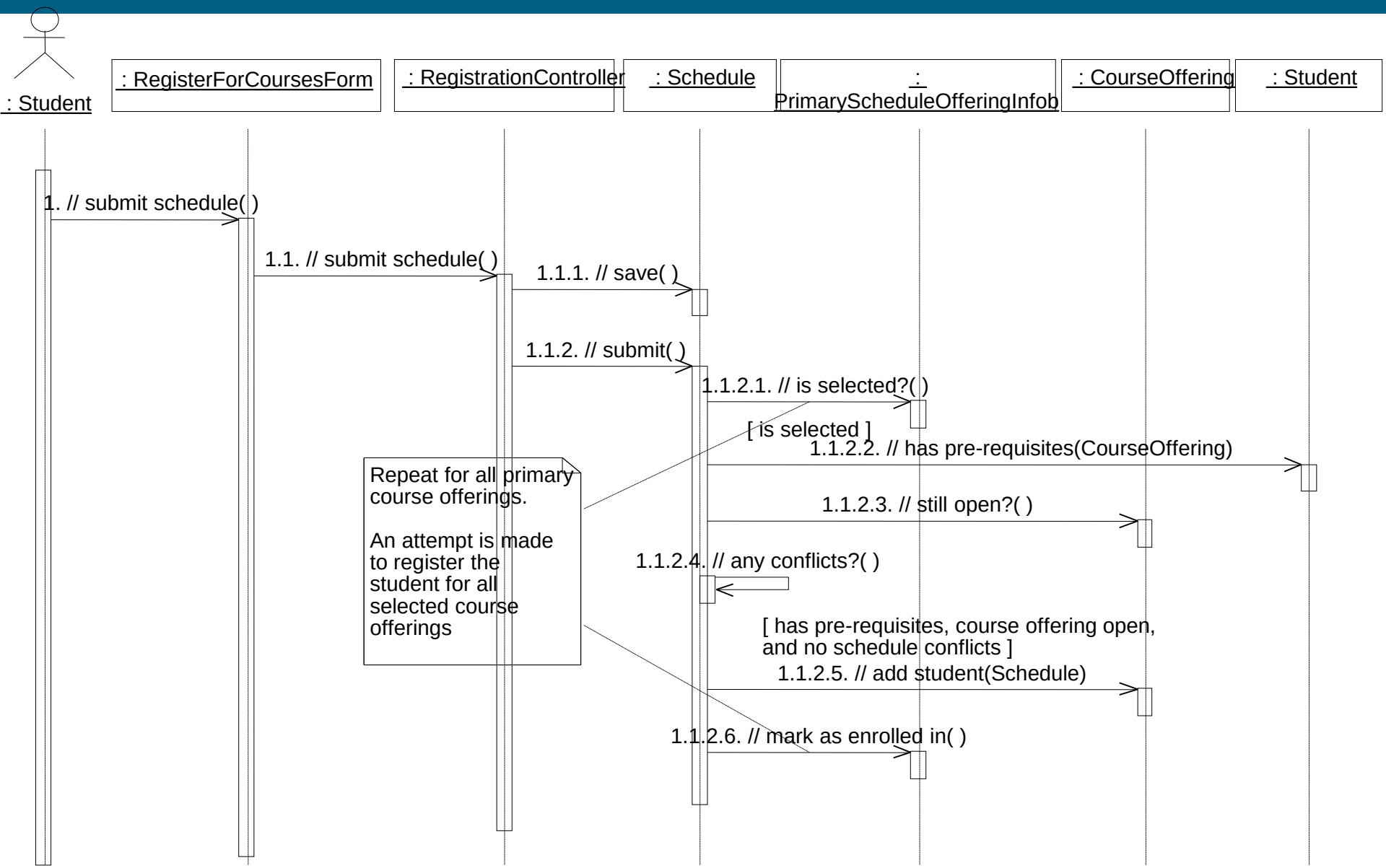
Sequence Diagram Example



Sequence Diagram Example



Sequence Diagram Example



Sequence Diagrams

1. Introduction
2. Elements of a Sequence Diagram
3. Guidelines for Creating Sequence Diagrams


QUESTIONS
ANSWERS

CONTENT

1. Activity Diagram
2. Sequence Diagram
3. Class Diagram
4. State Machine

Class Diagram

Class Diagram

1. Introduction
2. Structure Models
    1. Class, Attribute, and Operation
    2. Relationship
3. Object Identification
4. Class diagrams

INTRODUCTION

- A **structural model** describes the structure of the data that supports the business processes
- It illustrates people, places, or things about which information is captured and how they are related to one another
- The structure of data used in the system is represented through *class diagrams*, and *object diagrams*.

INTRODUCTION

- Purpose of Structural Models
  - Reduce the "semantic gap" between the real world and the world of software
  - Create a vocabulary for analysts and users
  - Represent things, ideas, and concepts of importance in the application domain

INTRODUCTION

- In UML:
  - **Structure Diagrams** show the **static structure** of the system and its parts on different abstraction and implementation **levels** and how they are related to each other.
  - The elements in a structure diagram represent the meaningful concepts of a system, and may include abstract, real world and implementation concepts.
  - Class Diagram, Object Diagram, Package Diagram, Composite Structure Diagram, Component Diagram, Deployment Diagram, Profile Diagram.

Class Diagram

1. Introduction
2. Structure Models
    1. Class, Attribute, and Operation
    2. Relationship
3. Object Identification
4. Class diagrams

Structure Models

- Structural model does contain analysis classes, attributes, operations, and the relationships among the analysis classes
- The structural model at this point should represent the responsibilities of each class and the collaborations among the classes
- Typically, structural models are depicted using class diagrams, and, in some cases, object diagrams

Structure Models

1. Class: template to define specific instances or objects
2. Object: instantiation of a class
3. Attributes: describes the object
4. Behaviours: specify what an object can do
5. Relationships

Structure Models

Class

- Template to define specific instances or objects
  - Concrete (can have real instances): employee, customer
  - Abstract (only exists to hold subclasses): person
- Typical Example
  - Application domain
  - user-interface, data structure, file structure, operating environment, document, and multimedia classes
- Example: Customer, Patient, Doctor, Appointment, Symptom…

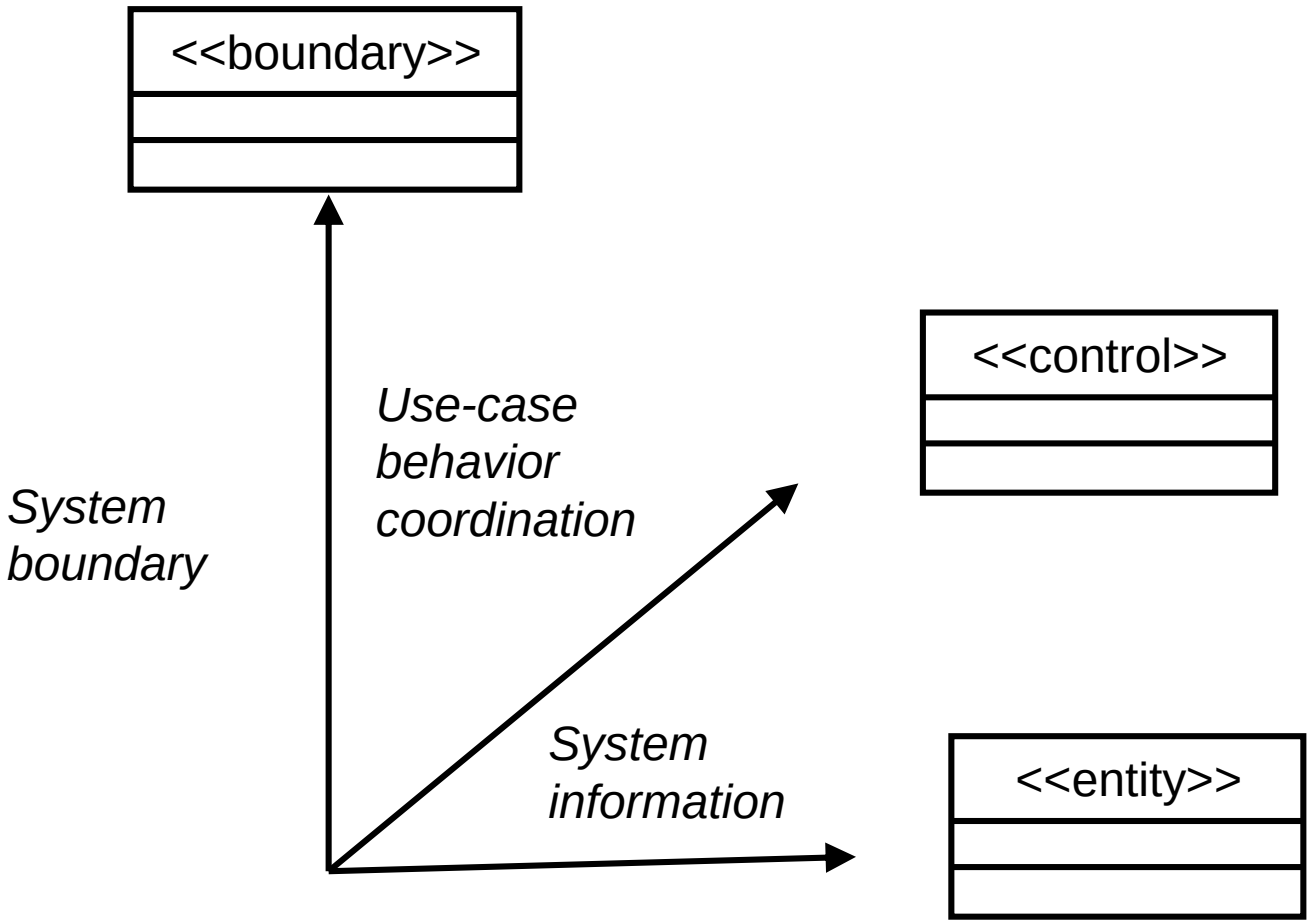
Structure Models

Class

- Nonstandard UML class
  1. Boundary class: user interface screen, system interface or device interface object.
  2. Control class: flow of control or behavior
  3. Entity class: information or data. Course, Teacher, CourseGrade …

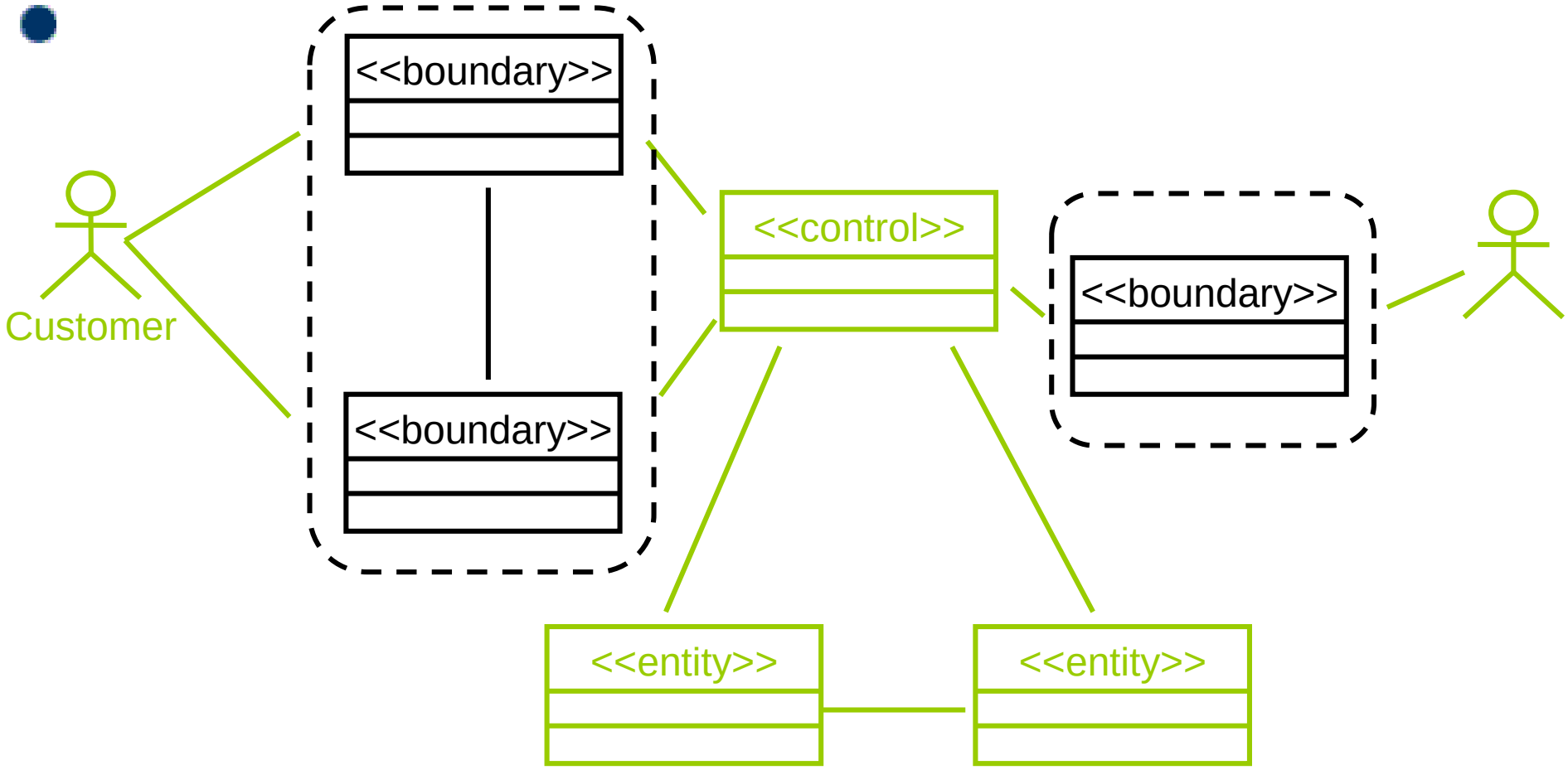
Structure Models

Class



Structure Models

Class



Structure Models

Attribute

- Units of information relevant to the description of the class
- Only attributes important to the task should be included
- Example:
  - Customer class: Name, Address, Phone number
  - Patient class: Name, Address, Phone, Insurance Carrier

Structure Models

Attribute

- Derived attributes: can be calculated from others
  - Age: calculated from birth date and current date
  - Overall score: calculated from midterm score, final score, homework score, attendance score, and bonus score
- Visibility
  - + Public (not hidden)
  - # Protected (hidden from all except immediate subclasses)
  - - Private (hidden from all other classes)

Classes and objects

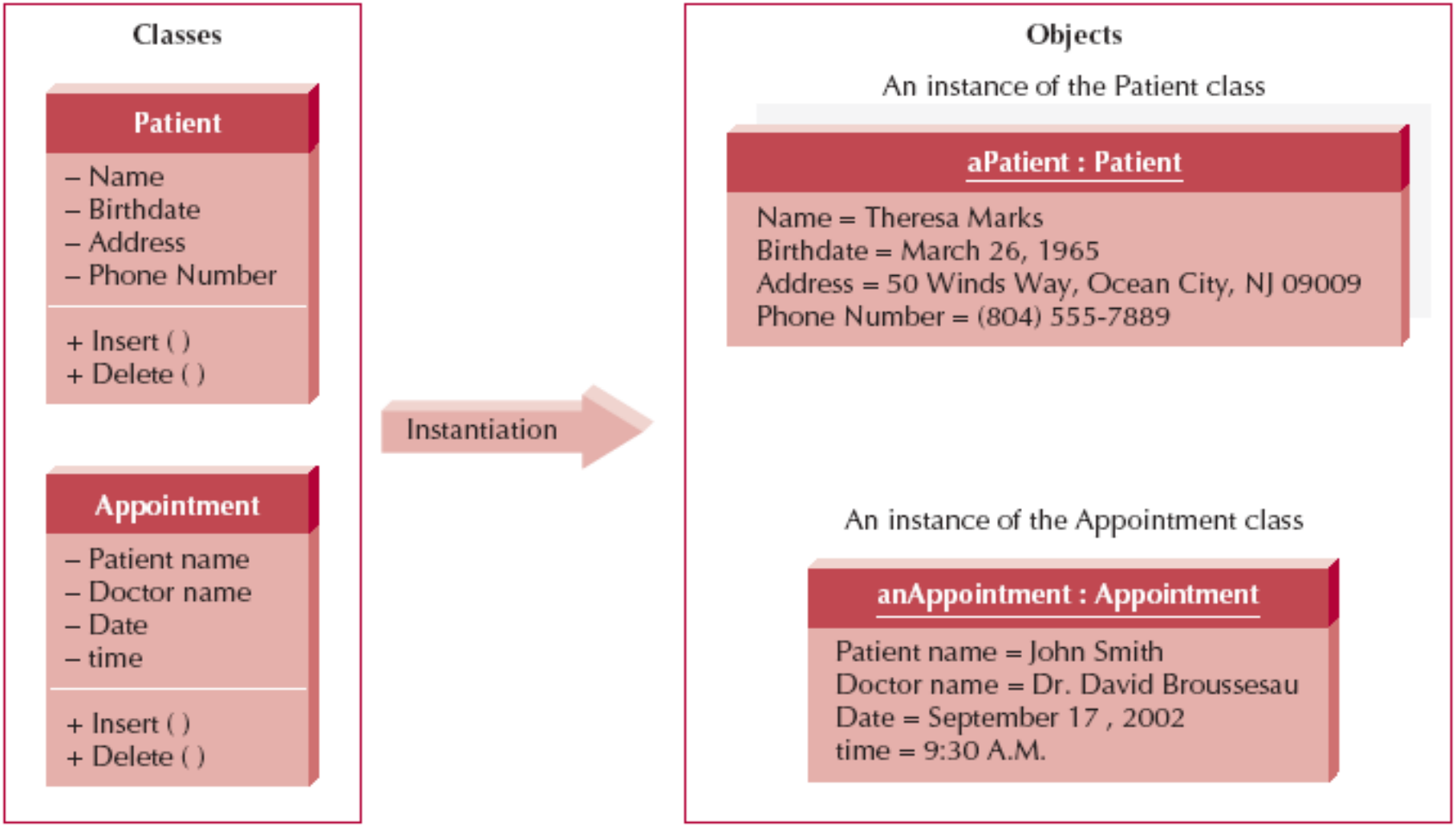


FIGURE 2-1 Classes and Objects

Structure Models

Behaviour (Operation/Method)

- Action that instances/objects can take
- Focus on relevant problem-specific operations (at this point)
- Example:
  - Patient object: make appointment, view appointment history
  - Student object: register for courses, check grade

Structure Models

Behaviour (Operation/Method)

- Constructor
  - Creates object
- Destructor
  - Removes object
- Query
  - Makes information about state available
- Update
  - Changes values of some or all attributes

Structure Models

Relationship

- *Generalization*
  - Enables *inheritance* of attributes and operations [...is a kind of...]
- *Aggregation*
  - Relates *parts* to the whole [..is a part of..]
- *Association*
  - Miscellaneous relationships between classes

Class Diagram

1. Introduction
2. Structure Models
    1. Class, Attribute, and Operation
    2. Relationship
3. Object Identification
4. Class diagrams

Object Identification

1. Textual Analysis
2. Brainstorming
3. Common object lists
4. Patterns
5. Combination of above techniques

Object Identification

Textual Analysis

- ✓ Reviewing the use-case diagrams and examining the text in the use-case descriptions to identify potential objects, attributes, operations, and relationships.
- ✓ Nouns suggest possible classes
- ✓ Verbs suggest possible operations

Object Identification

Brainstorming

- ✓ Individuals suggest potential classes that could be useful for the problem under consideration.
- ✓ It simply asks the participants to identify the objects based on their past experiences.

Object Identification

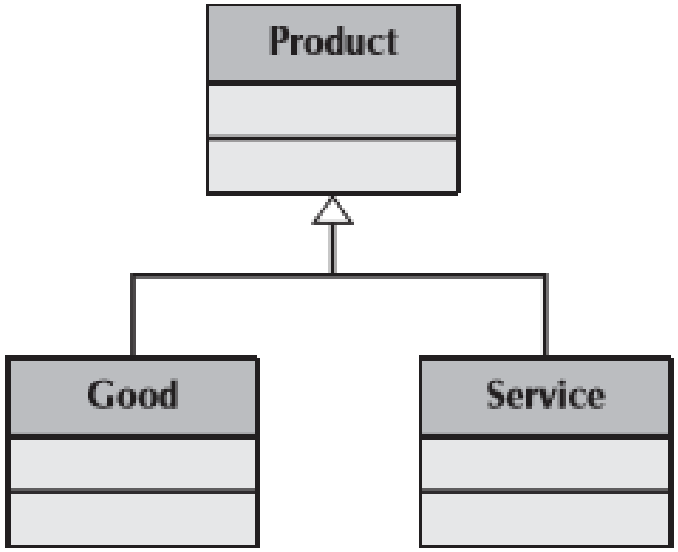
Common object lists

- ✓ List of objects common to the business domain of the system
- ✓ Reviewing the use cases can identify the roles that the people play in the problem, such as doctor, nurse, patient, or receptionist
- ✓ Places, containers, organizations, business records, catalogs, and policies

Object Identification

Patterns

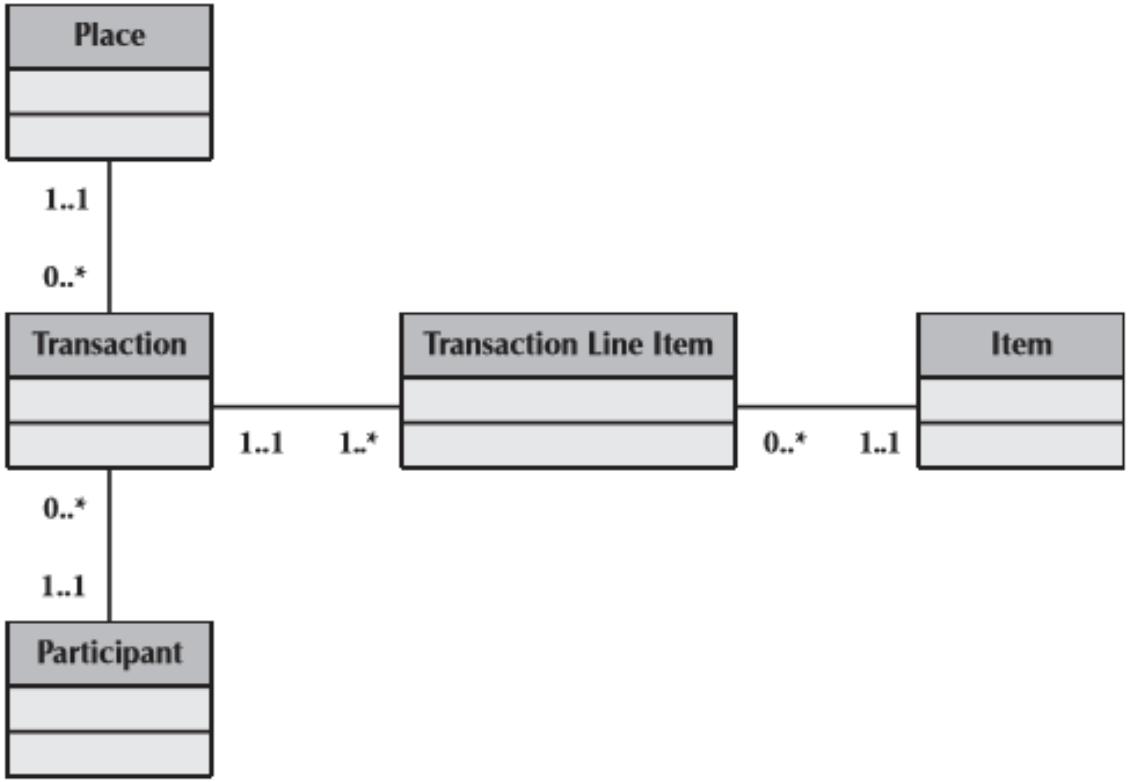
- ✓ a useful group of collaborating classes that provide a solution to a commonly occurring problem



Object Identification

Patterns

- ✓ a useful group of collaborating classes that provide a solution to a commonly occurring problem



Class Diagram

1. Introduction
2. Structure Models
    1. Class, Attribute, and Operation
    2. Relationship
3. Object Identification
4. Class diagrams

Class diagram

1. Elements of a class diagram
2. Examples

Class diagram

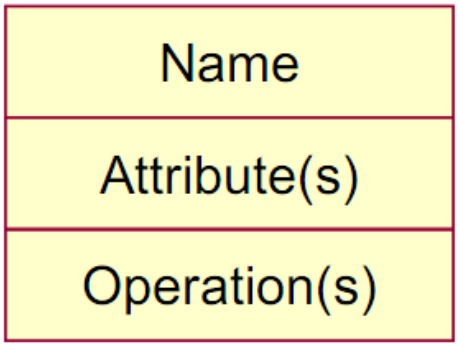
Elements of a class diagram

1. Class
2. Attribute
3. Operation
4. Association
5. Generalization
6. Aggregation
7. Composition

Class diagram

Elements of a class diagram

1. Class: represents a kind of person, place, or thing about which the system will need to capture and store information
2. Attribute: represents properties that describe the state of an object;
3. Operation: represents the actions or functions that a class can perform



Class diagram

Elements of a class diagram

4. Association:

- ✓ Represents a relationship between multiple classes or a class and itself
- ✓ Contains multiplicity symbols, which represent the minimum and maximum times a class instance can be associated with the related class instance

Class diagram

Elements of a class diagram

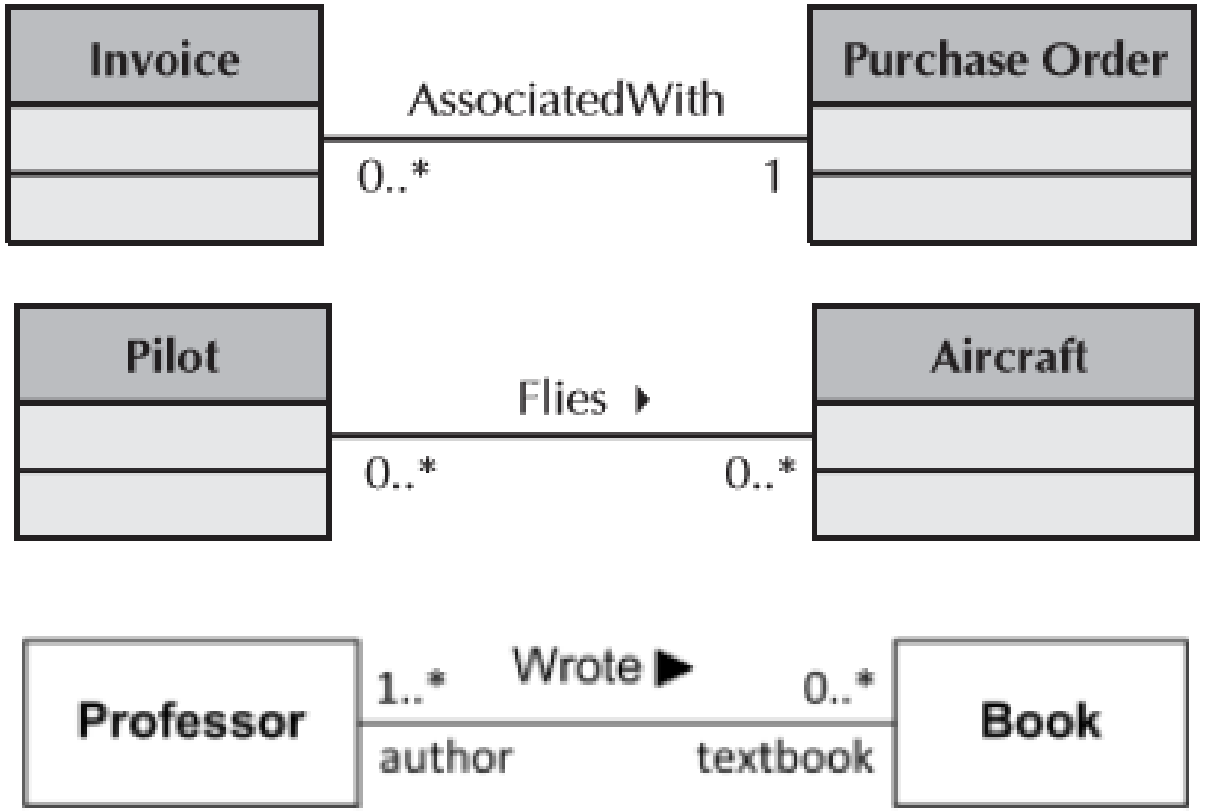
4. Association:

- ✓ Represents a relationship between multiple classes or a class and itself
- ✓ Contains multiplicity symbols, which represent the minimum and maximum times a class instance can be associated with the related class instance

Class diagram

Elements of a class diagram

4. Association:



Class diagram

Elements of a class diagram

4. Association:

Exactly one	1		A department has one and only one boss.
Zero or more	0..*		An employee has zero to many children.
One or more	1..*		A boss is responsible for one or more employees.

Class diagram

Elements of a class diagram

4. Association:

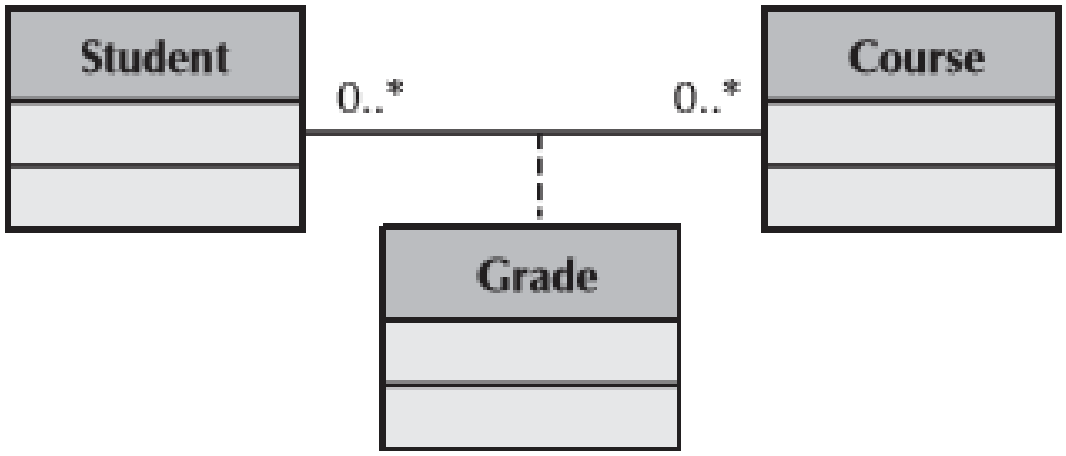
Zero or one	0..1	Employee — 0..1 — Spouse	An employee can be married to zero or one spouse.
Specified range	2..4	Employee — 2..4 — Vacation	An employee can take from two to four vacations each year.
Multiple, disjoint ranges	1..3,5	Employee — 1..3,5 — Committee	An employee is a member of one to three or five committees.

Class diagram

Elements of a class diagram

4. Association:

Association Classes

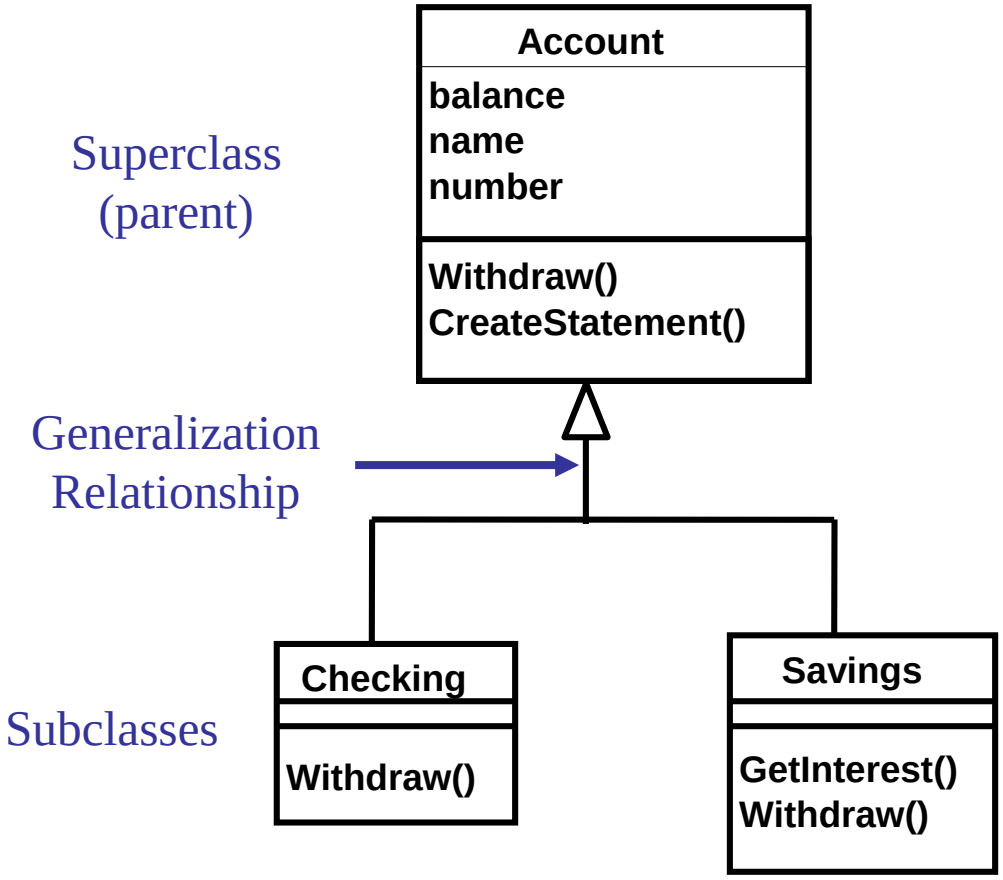


Class diagram

Elements of a class diagram

5. Generalization :

✓ Represents a-kind-of relationship between multiple classes

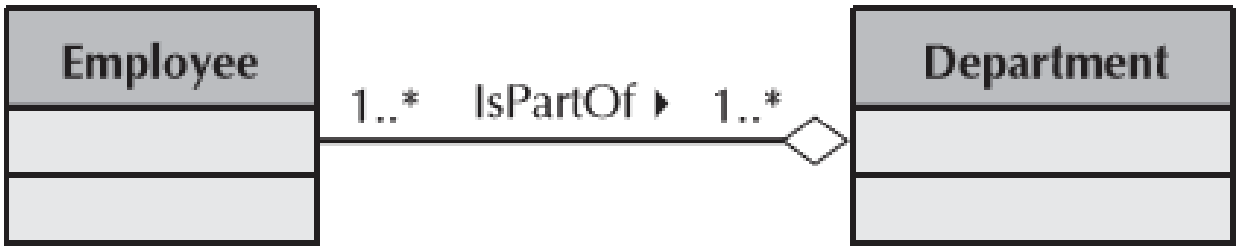


Class diagram

Elements of a class diagram

6. Aggregation:

- ✓ Represents a logical a-part-of relationship between multiple classes or a class and itself
- ✓ Is a special form of an association

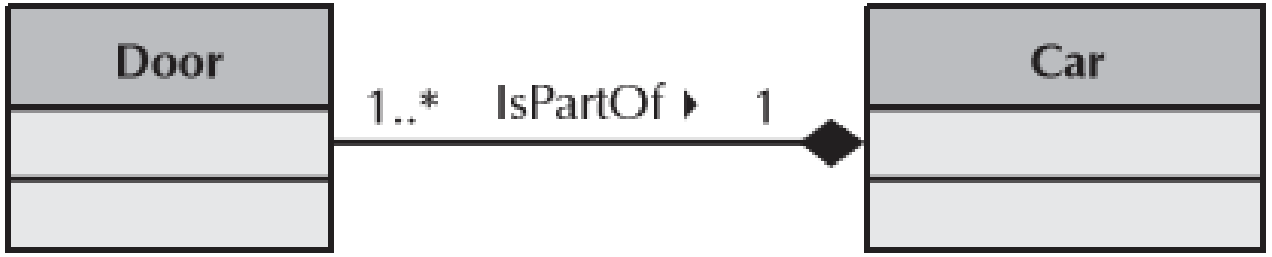


Class diagram

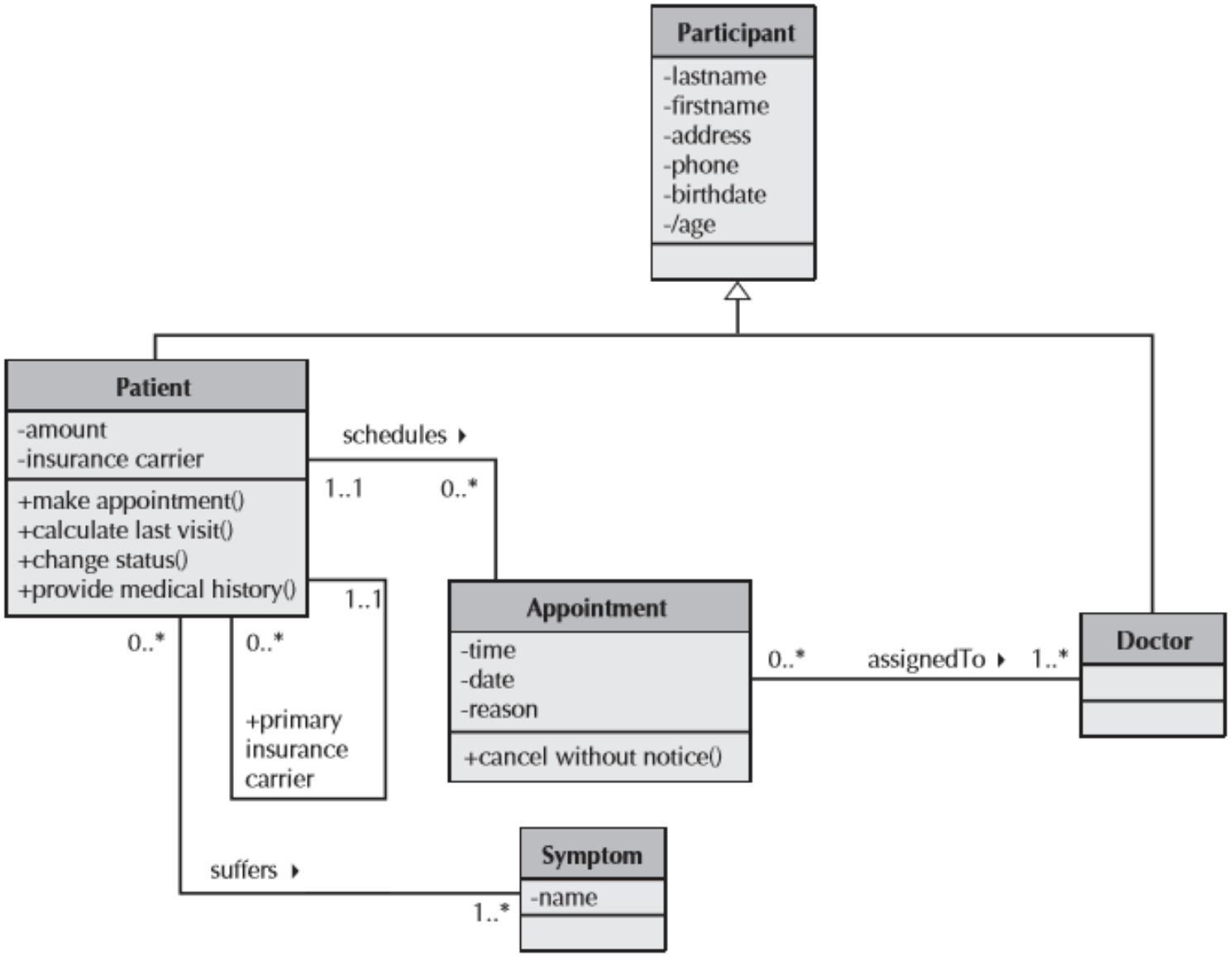
Elements of a class diagram

7. Composition:

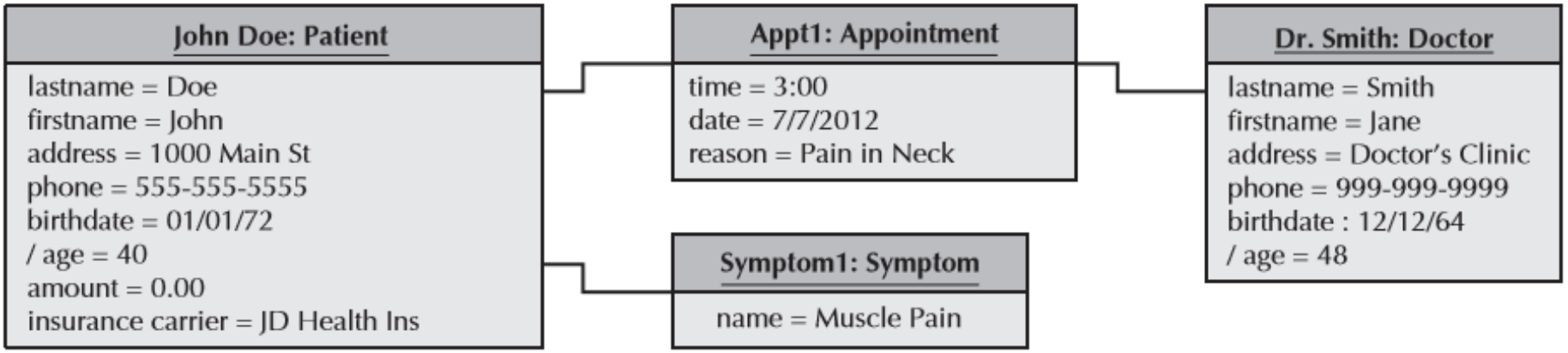
- ✓ Represents a physical a-part-of relationship between multiple classes or a class and itself
- ✓ It is a *whole/part* relationship.

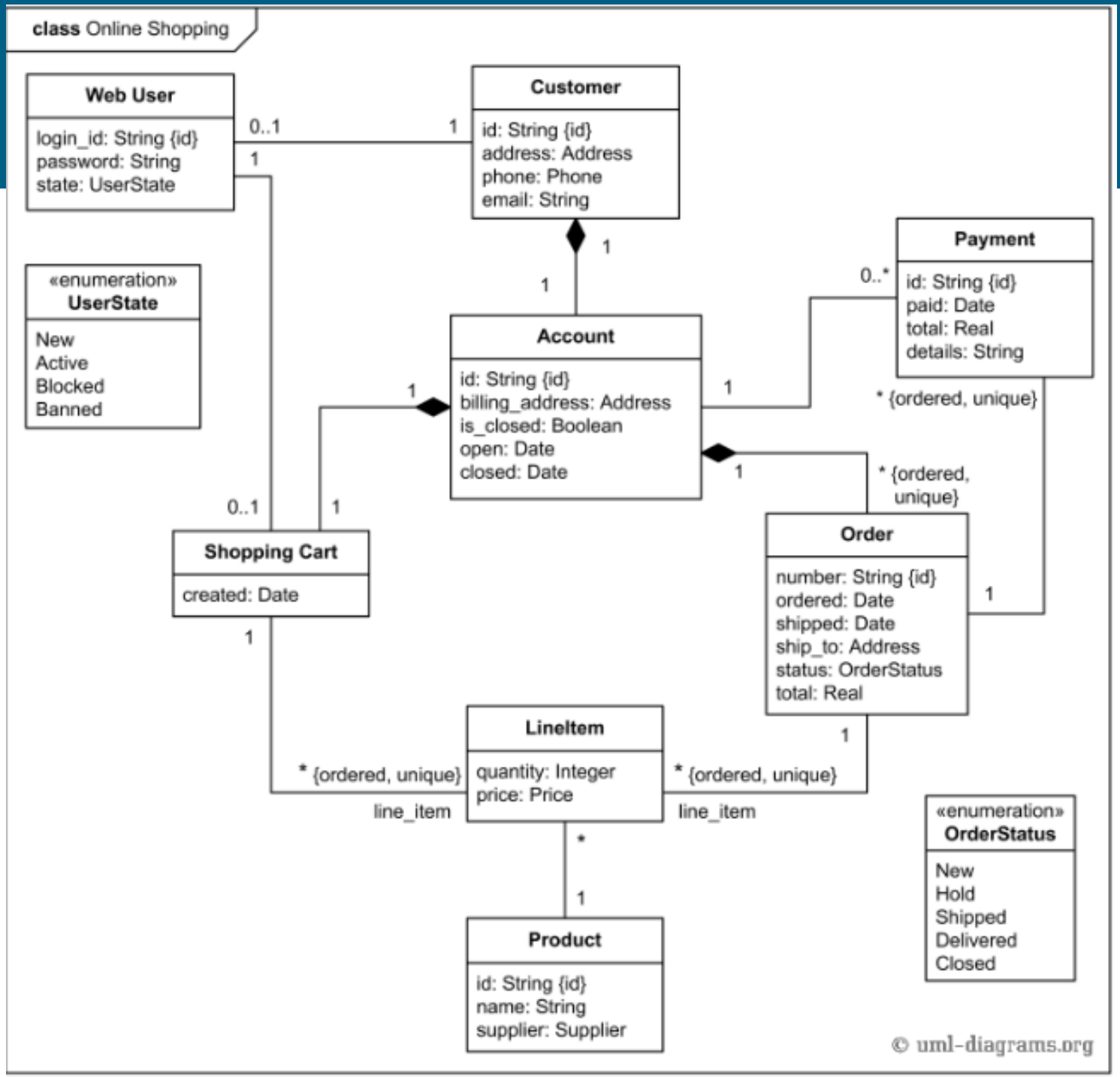


Class diagram



Object diagram




class Online Shopping
Web User
login_id: String {id}
password: String
state: UserState
0..1
1
Customer
id: String {id}
address: Address
phone: Phone
email: String
1
1
«enumeration»
UserState
New
Active
Blocked
Banned
Payment
0..*
id: String {id}
paid: Date
total: Real
details: String
Account
id: String {id}
billing_address: Address
is_closed: Boolean
open: Date
closed: Date
1
1
1
1
* {ordered, unique}
* {ordered, unique}
0..1
1
Shopping Cart
created: Date
Order
number: String {id}
ordered: Date
shipped: Date
ship_to: Address
status: OrderStatus
total: Real
1
1
1
LineItem
quantity: Integer
price: Price
* {ordered, unique}
line_item
* {ordered, unique}
line_item
*
1
Product
id: String {id}
name: String
supplier: Supplier
«enumeration»
OrderStatus
New
Hold
Shipped
Delivered
Closed
© uml-diagrams.org


QUESTIONS
ANSWERS

CONTENT

1. Activity Diagram
2. Sequence Diagram
3. Class Diagram
4. State Machine

State Machines

State Machines

1. Introduction
2. Elements of a State Machines
3. Guidelines for Creating State Machines

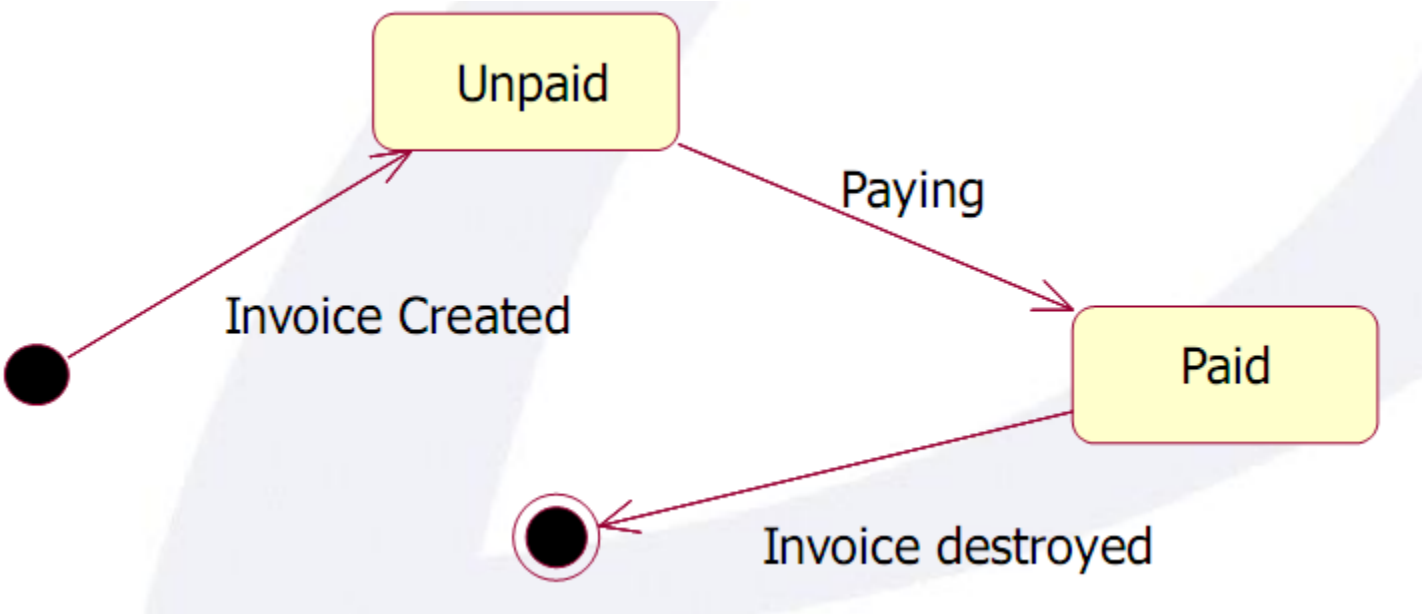
State Machines

Introduction

- ✓ A state machine shows the different states through which a single object passes during its life.
- ✓ Typically, state machines are not used for all objects, they are used with complex objects to help simplify the design of algorithms for their methods
- ✓ The behavioral state machine shows the different states of the object and what events cause the object to change from one state to another

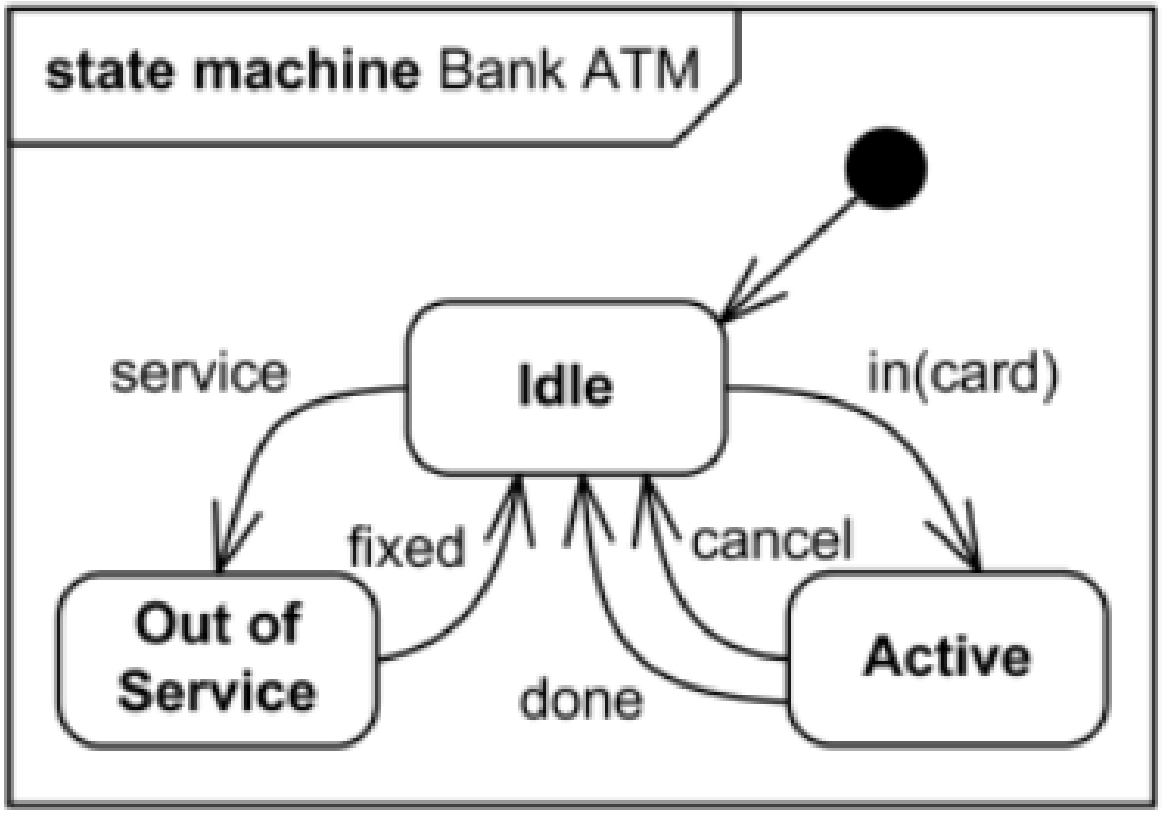
State Machines

Introduction



State Machines

Introduction



High level behavioral state machine for bank ATM

State Machines

Elements of a State Machines

1. States
2. An initial state, A final state
3. Events
4. Transitions
5. Actions
6. Activities
7. Frame

State Machines

Elements of a State Machines

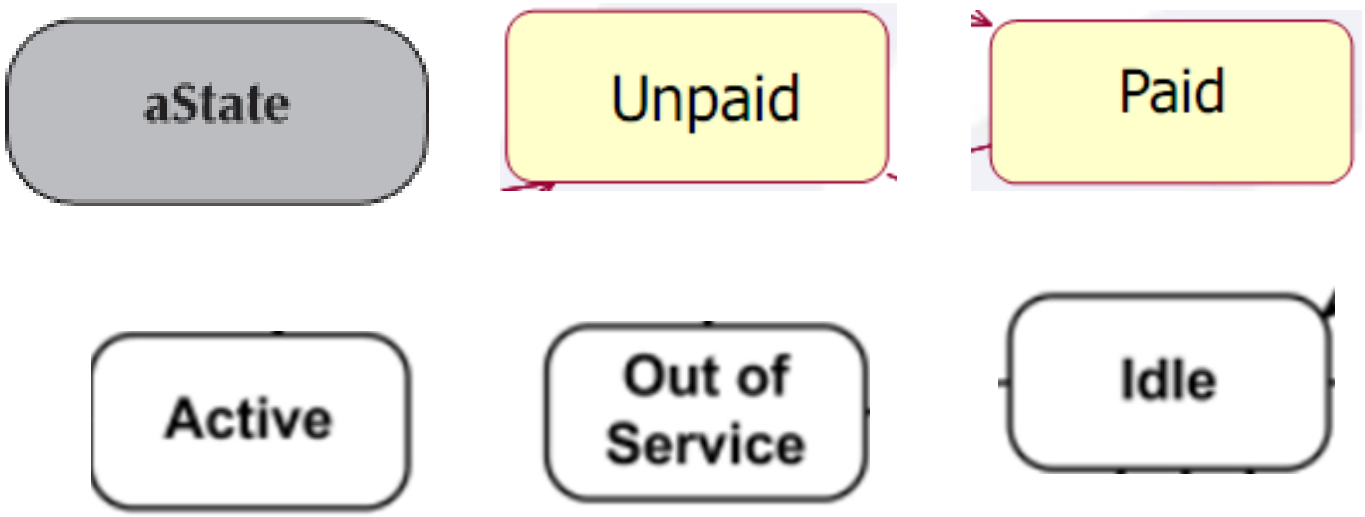
1. States
2. An initial state, A final state
3. Events
4. Transitions
5. Actions
6. Activities
7. Frame

State Machines

Elements of a State Machines

1. State

- ✓ Has a name that represents the state of an object
- ✓ Is shown as a rectangle with rounded corners.



State Machines

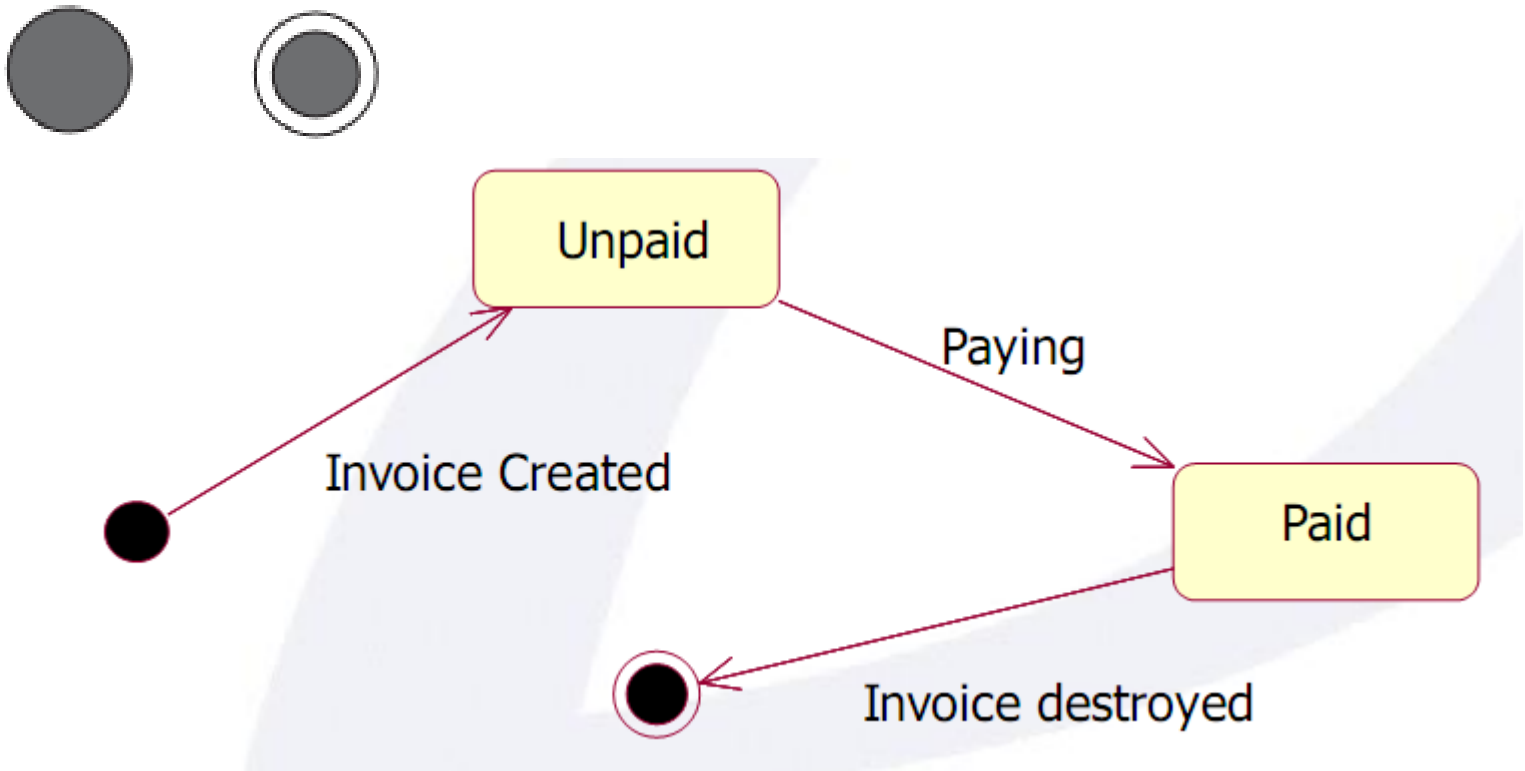
Elements of a State Machines

1. States
2. An initial state, A final state
3. Events
4. Transitions
5. Actions
6. Activities
7. Frame

State Machines

Elements of a State Machines

2. An initial state, A final state



State Machines

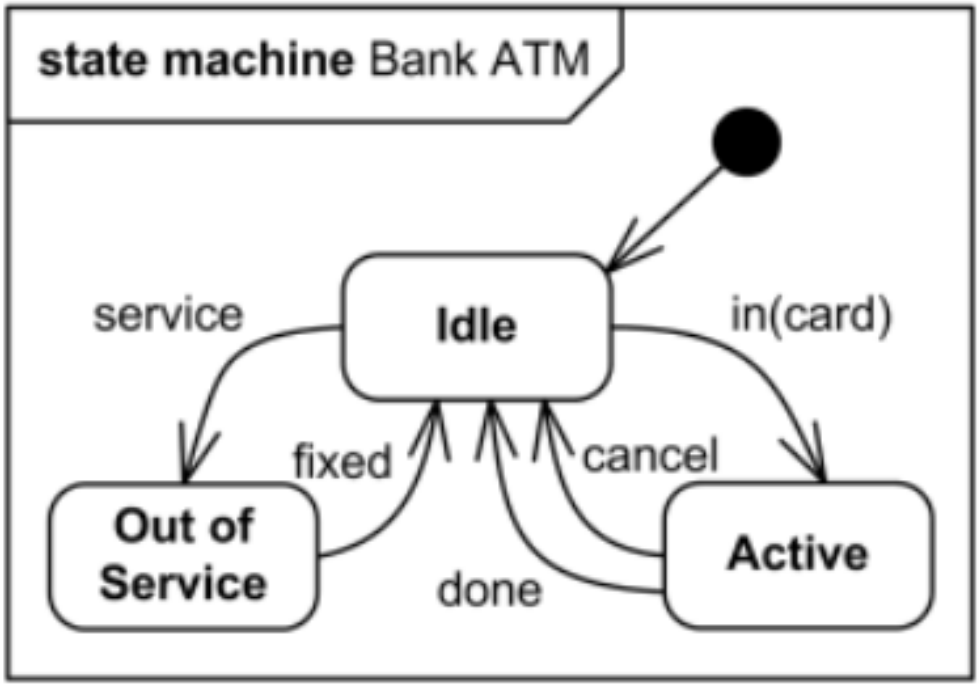
Elements of a State Machines

1. States
2. An initial state, A final state
3. Events
4. Transitions
5. Actions
6. Activities
7. Frame

State Machines

Elements of a State Machines

7. Frame: indicates the context of the State Machine



High level behavioral state machine for bank ATM

State Machines

Elements of a State Machines

1. States
2. An initial state, A final state
3. Events
4. Transitions
5. Actions
6. Activities
7. Frame

State Machines

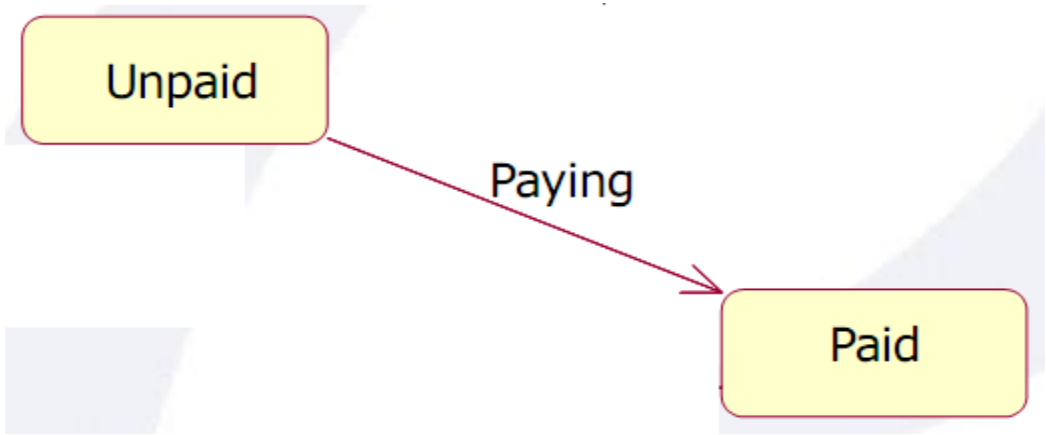
Elements of a State Machines

3. Event

- ✓ Is an occurrence that triggers a change in state
- ✓ Is used to label a transition

4. Transition

- ✓ Indicates that an object in the first state will enter the second state



State Machines

Elements of a State Machines

1. States
2. An initial state, A final state
3. Events
4. Transitions
5. Actions
6. Activities
7. Frame

State Machines

Elements of a State Machines

5. Action

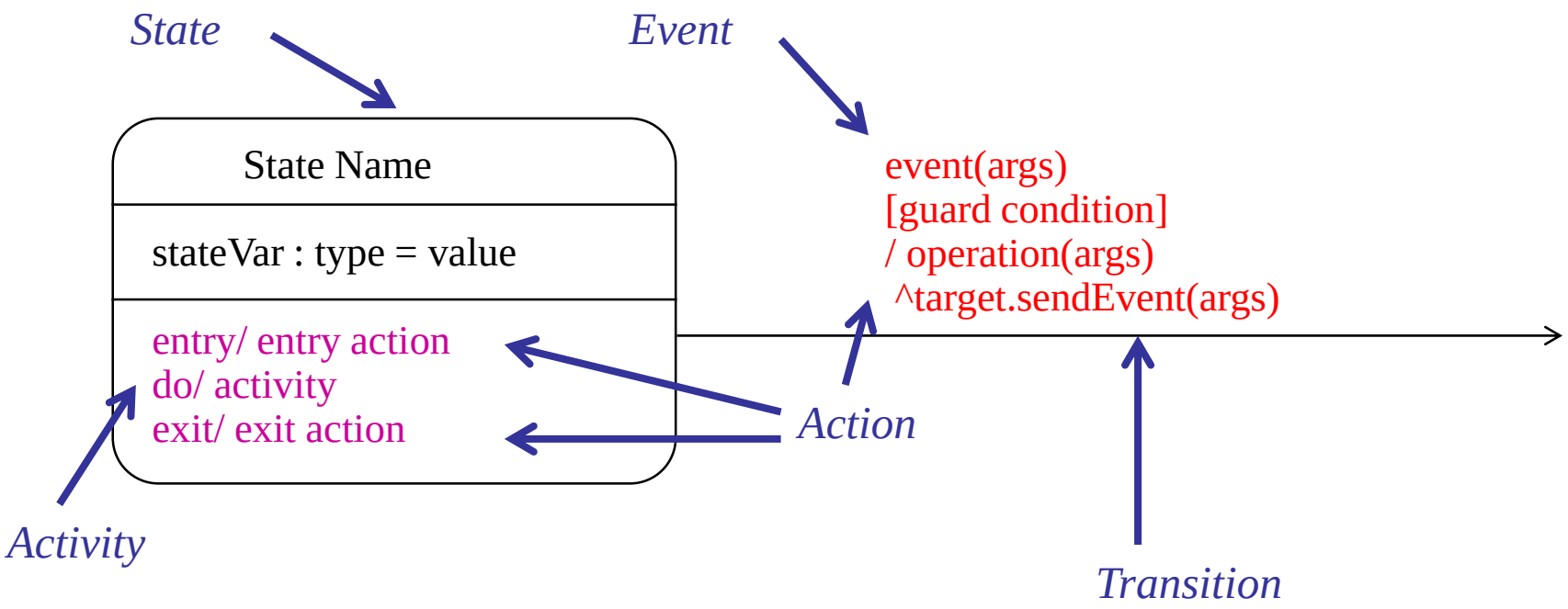
- ✓ Action cannot be interrupted
- ✓ It is associated with a transition

6. Activity

- ✓ Activity can be interrupted. Activities take a long period of time to complete
- ✓ It can be started and stopped by an action.

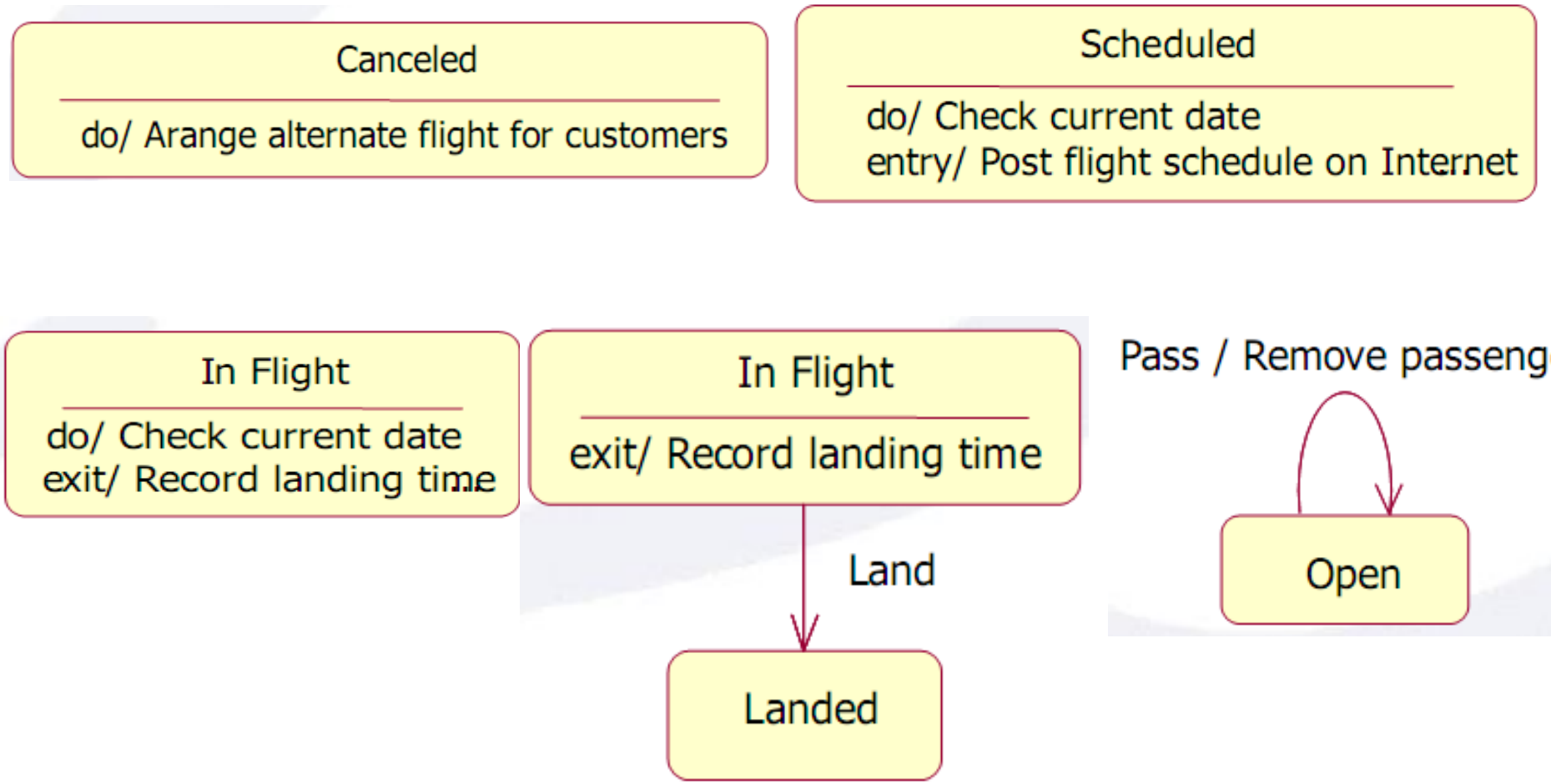
State Machines

Elements of a State Machines

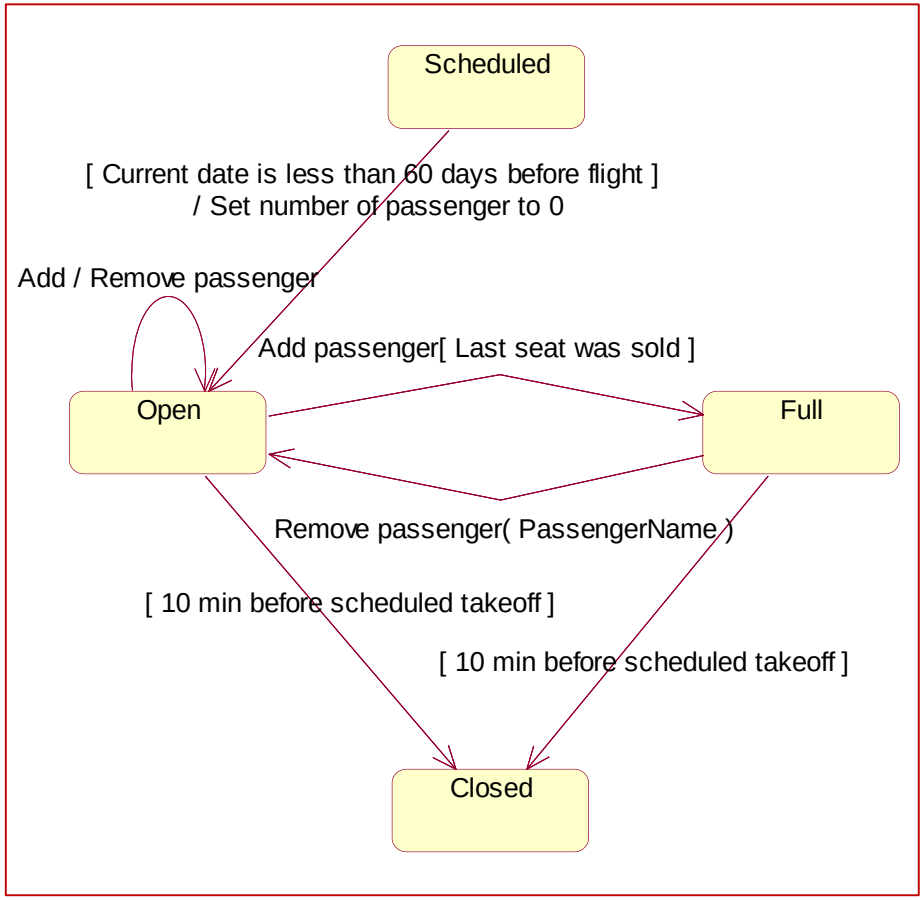


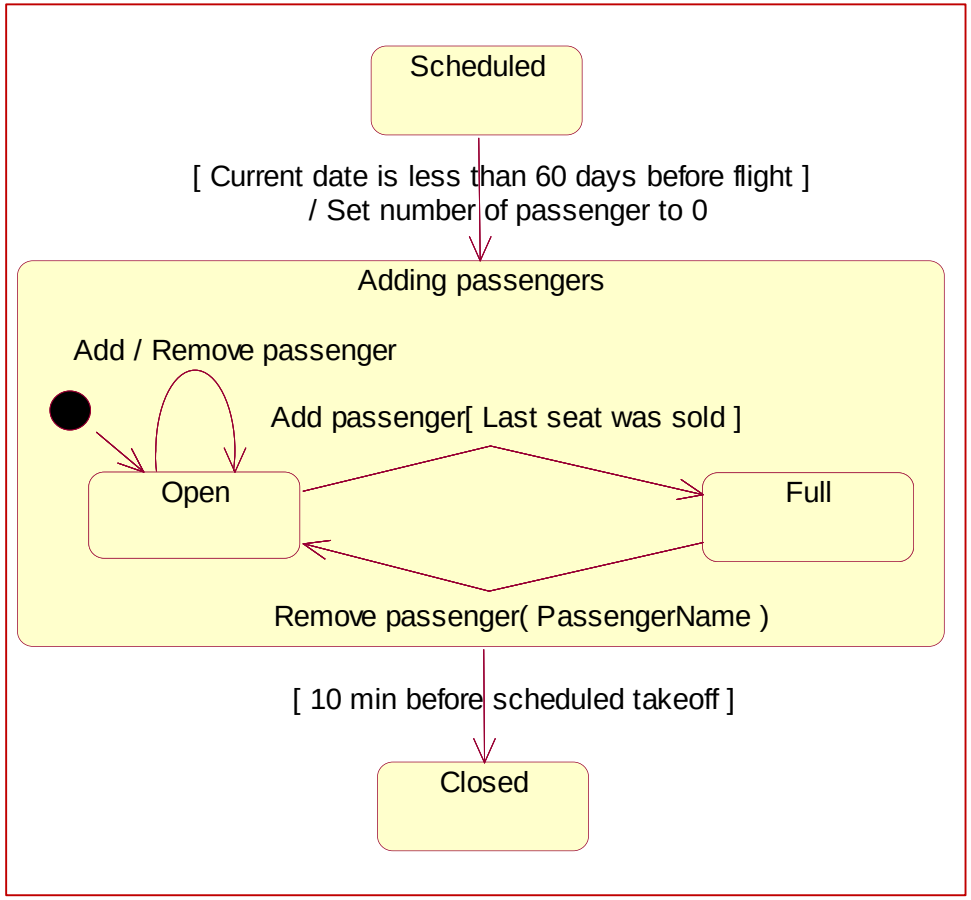
State Machines

Elements of a State Machines



State Machines





State Machines

Creating a State Machines

1. Set context
2. Identify object states
3. Layout diagram
4. Add transition
5. Validate

CONTENT

1. Activity Diagram
2. Sequence Diagram
3. Class Diagram
4. State Machine

Reference

1. Kendall & Kendall, *Systems Analysis and Design*, 9th edition, Prentice Hall, 2014.
2. Alan Dennis, Babara Haley Wixom, David Tegarden, *Systems Analysis and Design: An Object-Oriented Approach with UML, 5th Edition*, Wiley, 2015.
3. *System Analysis and Design with UML slides*, Faculty of Information Systems, University of Information Technology, HCMC, 2018.
4. https://www.uml-diagrams.org/sequence-diagrams.html


QUESTIONS
ANSWERS