

Behavioral Modeling



- ❑ Behavioral models describe the **internal dynamic aspects** of an information system that supports business processes in an organization
- ❑ Key UML behavioral models are: **sequence diagrams** and **behavioural state machines**

Objectives

- Understand the **rules and style guidelines** for sequence diagrams and behavioral state machines.
- Understand the processes used to create sequence diagrams and behavioral state machines.
- Be able to create sequence diagrams and behavioral state machines.
- Understand the relationship between the **behavioral** models and the **structural** and **functional** models.

Purpose of Behavioral Models



- ❑ Show how objects collaborate to support each use case in the structural model
- ❑ Depict the internal view of the business process
- ❑ To show the effects of varied processes on the system

Interaction Diagram Components



- ☑ **Objects**

- Instance of a class

- ☑ **Operations**

- Send and receive messages

- ☑ **Messages**

- Tell object to execute a behavior

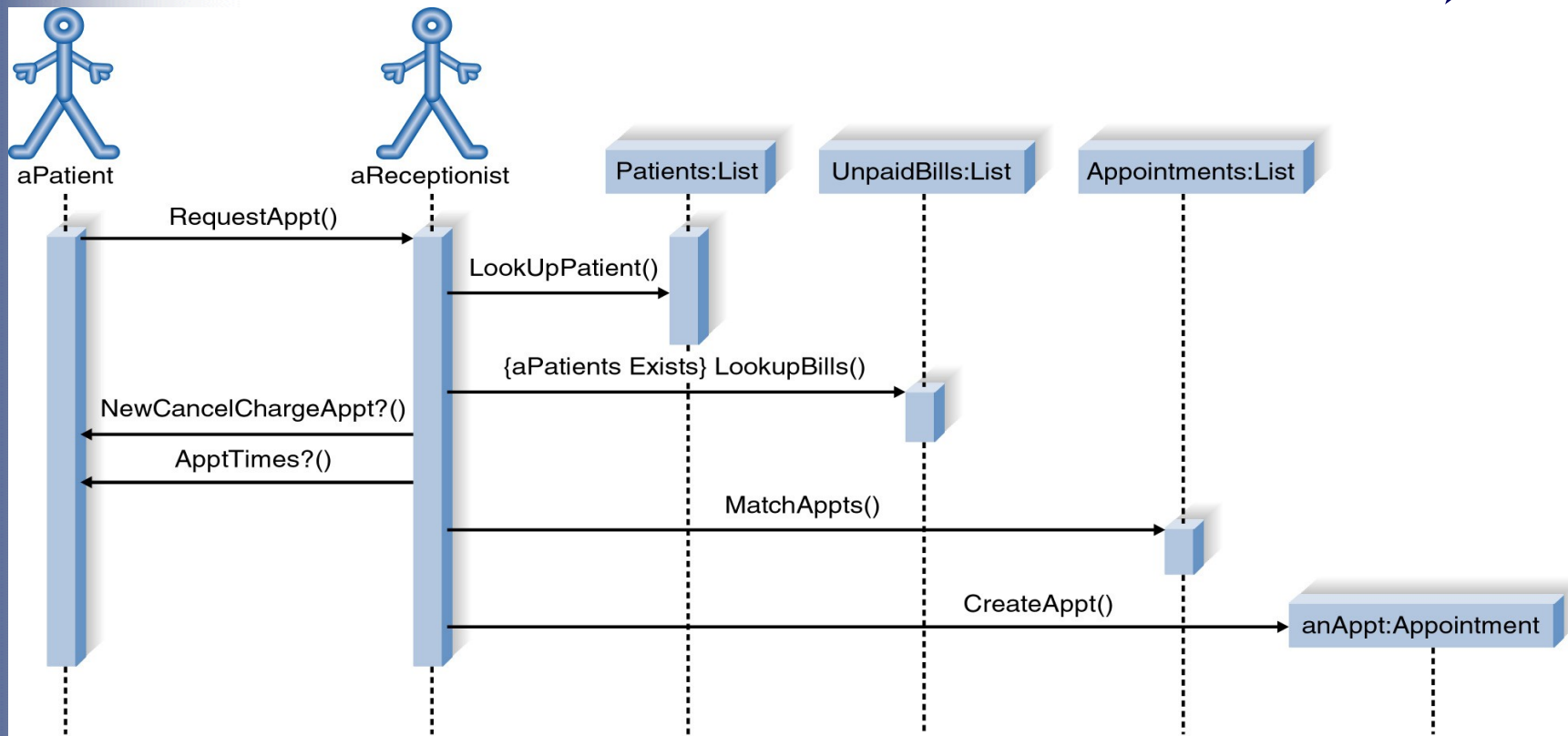
Sequence Diagrams



- ❑ Illustrate the **objects** that participate in a use-case
- ❑ Show the **messages** that pass between objects **for a particular use-case**

Example Sequence Diagram

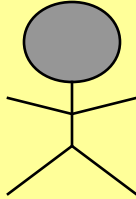





Make Appointment



Dennis: SAD

Fig: 8-1 W-30 100% of size
Fine Line Illustrations (516) 501-0400

Sequence Diagram Syntax

ACTOR	
OBJECT	
LIFELINE	
EXECUTION OCCURRENCE (FOCUS OF CONTROL)	
MESSAGE	
OBJECT DESTRUCTION	

Building a Sequence Diagram



1. Determine the **context** of the sequence diagram
2. Identify the participating **objects**
3. Set the **lifeline for each object**
4. Add **messages**
5. Place the **execution occurrence (focus of control)** on each object's lifeline
6. **Validate** the sequence diagram


Use Case Name: Place Order	ID: 3	Importance Level: High
Primary Actor: Customer	Use Case Type: Detail, Essential	
Stakeholders and Interests: Customer—wants to search Web site to purchase CD. EM manager—wants to maximize customer satisfaction.		
Brief Description: This use case describes how customers can search the Web site and place orders.		
Trigger: Customer visits Web site and places order		
Type: External		
Relationships: Association: Customer Include: Checkout, Maintain Order Extend: Generalization:		
Normal Flow of Events: 1. Customer submits a search request to the system. 2. The System provides the Customer a list of recommended CDs. 3. The Customer chooses one of the CDs to find out additional information. 4. The System provides the Customer with basic information and reviews on the CD. 5. The Customer calls the Maintain Order use case. 6. The Customer iterates over 3 through 5 until done shopping. 7. The Customer executes the Checkout use case. 8. The Customer leaves the Web site.		
Subflows:		
Alternate/exceptional Flows: 3a-1. The Customer submits a new search request to the system. 3a-2. The Customer iterates over steps 2 through 3 until satisfied with search results or gives up. 7a. The Customer aborts the order.		

FIGURE 6-15 Places Order Use Case after Step 11

YOUR**6-6 Campus Housing****TURN**

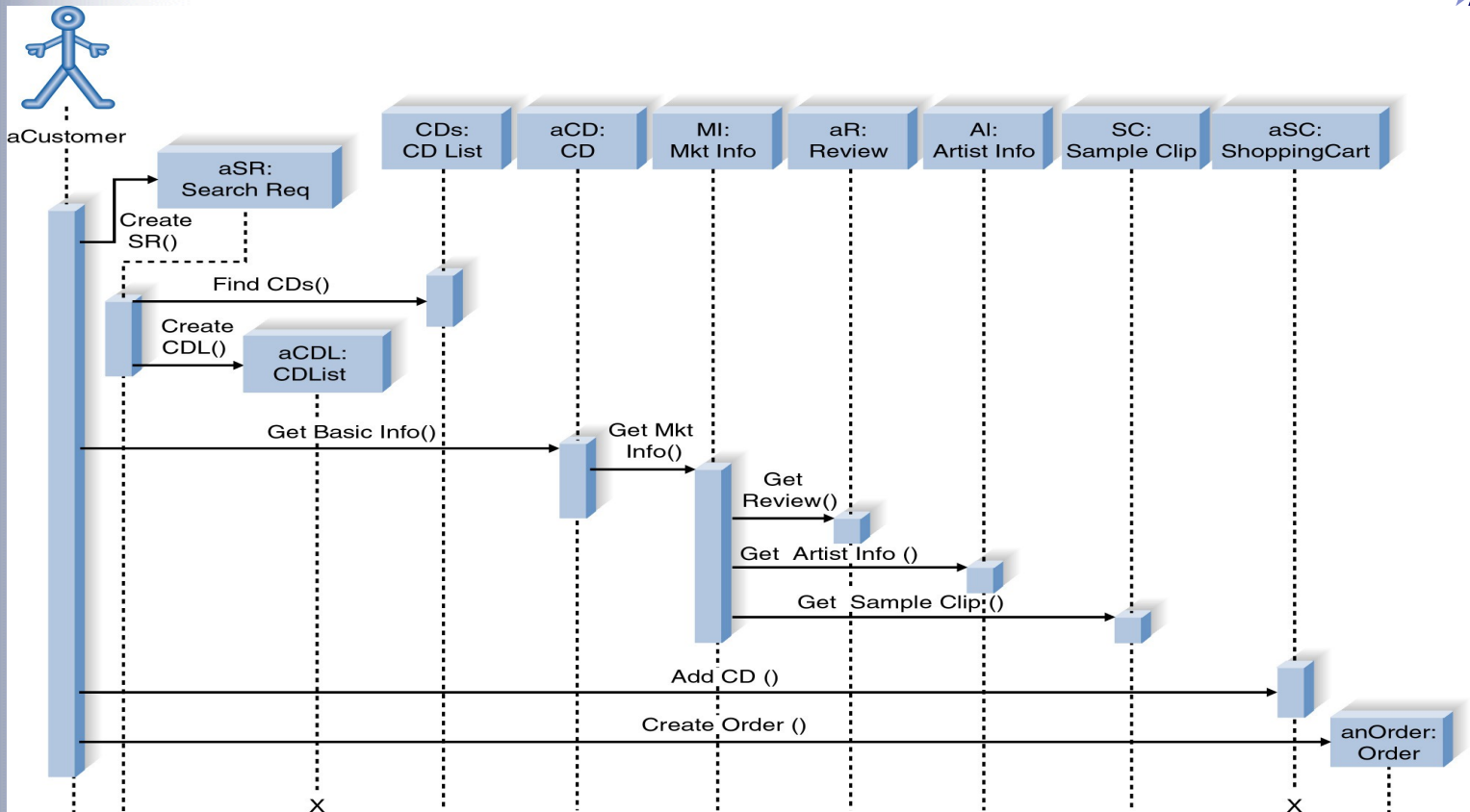
Create a set of use cases for the following high-level processes in a housing system run by the campus housing service. The campus housing service helps students find apartments. Apartment owners fill in information forms about the rental units they have available (e.g., location, number of bedrooms, monthly rent), which are then entered into a database. Students can search

through this database via the Web to find apartments that meet their needs (e.g., a two-bedroom apartment for \$400 or less per month within 1/2 mile of campus). They then contact the apartment owners directly to see the apartment and possibly rent it. Apartment owners call the service to delete their listing when they have rented their apartment(s).




Normal Flow of Events:

1. **Customer** submits a search request to the system.
2. The system provides the **customer** a list of recommended CDs.
3. The **customer** chooses one of the CDs to find additional information.
4. The system provides the **customer** with basic information &
CD Reviews
5. The **customer** calls the **maintain order use case**.
6. The **customer** iterates over 3 through 5 until finished shopping.
7. The **customer** executes the **checkout use case**.
8. The **customer** leaves the website.




Dennis: SAD
 Fig: 8-5 W-32 100% of size
 Fine Line Illustrations (516) 501-0400

Behavioral State Machines (State Chart Diagrams)



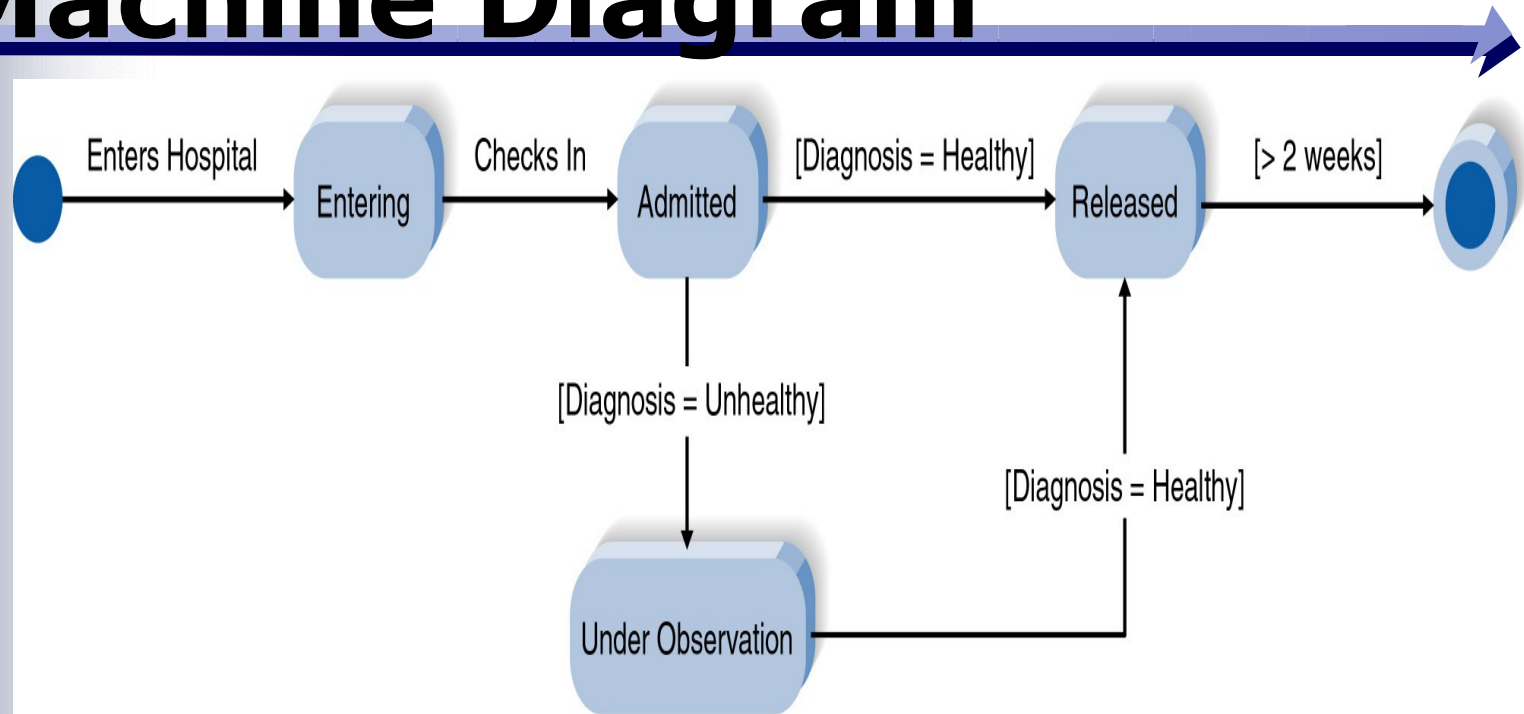
- ☑ The behavioral state machine is a **dynamic** model that shows the **different states** of the **object** and what **events** cause the **object to change from one state to another**, along with its responses and actions.

Elements of a Behavioral State Machine



- ❑ States (idle conditions)
- ❑ Events (triggers)
- ❑ Transitions (changes in state)
- ❑ Actions (cause transitions)
- ❑ Activities (groups of actions)

Example Behavioral State Machine Diagram








Dennis: SAD

Fig: 8-11 W-37 100% of size

Fine Line Illustrations (516) 501-0400

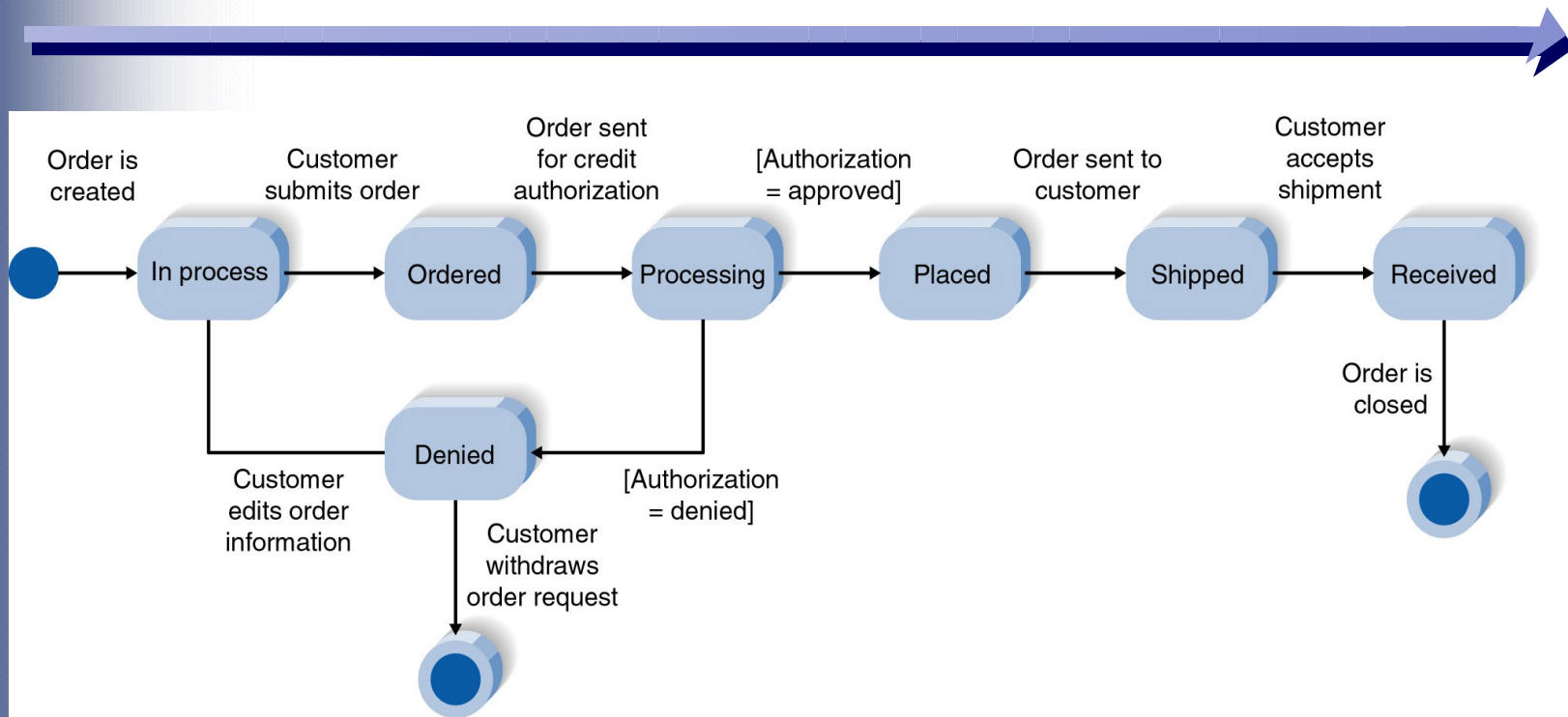
Behavioral State Machine Diagram Syntax

A STATE	
AN INITIAL STATE	
A FINAL STATE	
AN EVENT	anEvent
A TRANSITION	
A Frame	

Building Behavioral State Machine Diagrams



- ❑ Set the **context**
- ❑ Identify the **initial final**, and **stable states** of the object
- ❑ Determine **the order** in which the object will **pass through** stable states
- ❑ Identify the **events**, **actions**, and guard conditions associated with the transitions
- ❑ Validate the state machine diagram



Dennis: SAD
 Fig: 8-15 W-38a 100% of size
 Fine Line Illustrations (516) 501-0400

Summary



- ☒ *Sequence diagrams* illustrate the classes that participate in a use case and the messages that pass between them.
- ☒ *Behavioral State Machine diagrams* show the different states that a single class passes through in response to events.