BIT – 2015

Object Oriented Analysis and Design -IT 3105

Revision

OUTLINE OF SYLLABUS

Topics	Min. number of hours
• Introduction to Object Oriented Concepts	02
• Object Oriented Analysis and Modeling	01
• Software Development Process	02
• Creating Use Case Diagrams	05
• Identifying Classes ,Packages and drawing Class diagrams, Object Diagrams	06
• Object Oriented Design and Modeling using UML	04
• Working with State diagrams	03
• Discovering Object Interactions	05

OUTLINE OF SYLLABUS Cont…

Topics	Min. number of hours
• Working with Activity Diagrams	03
• Component and Deployment Diagrams	01
• New diagrams in UML 2.x , Model Driven Architecture (MDA), Executable UML	03
• Case Studies	10
Total	**45**

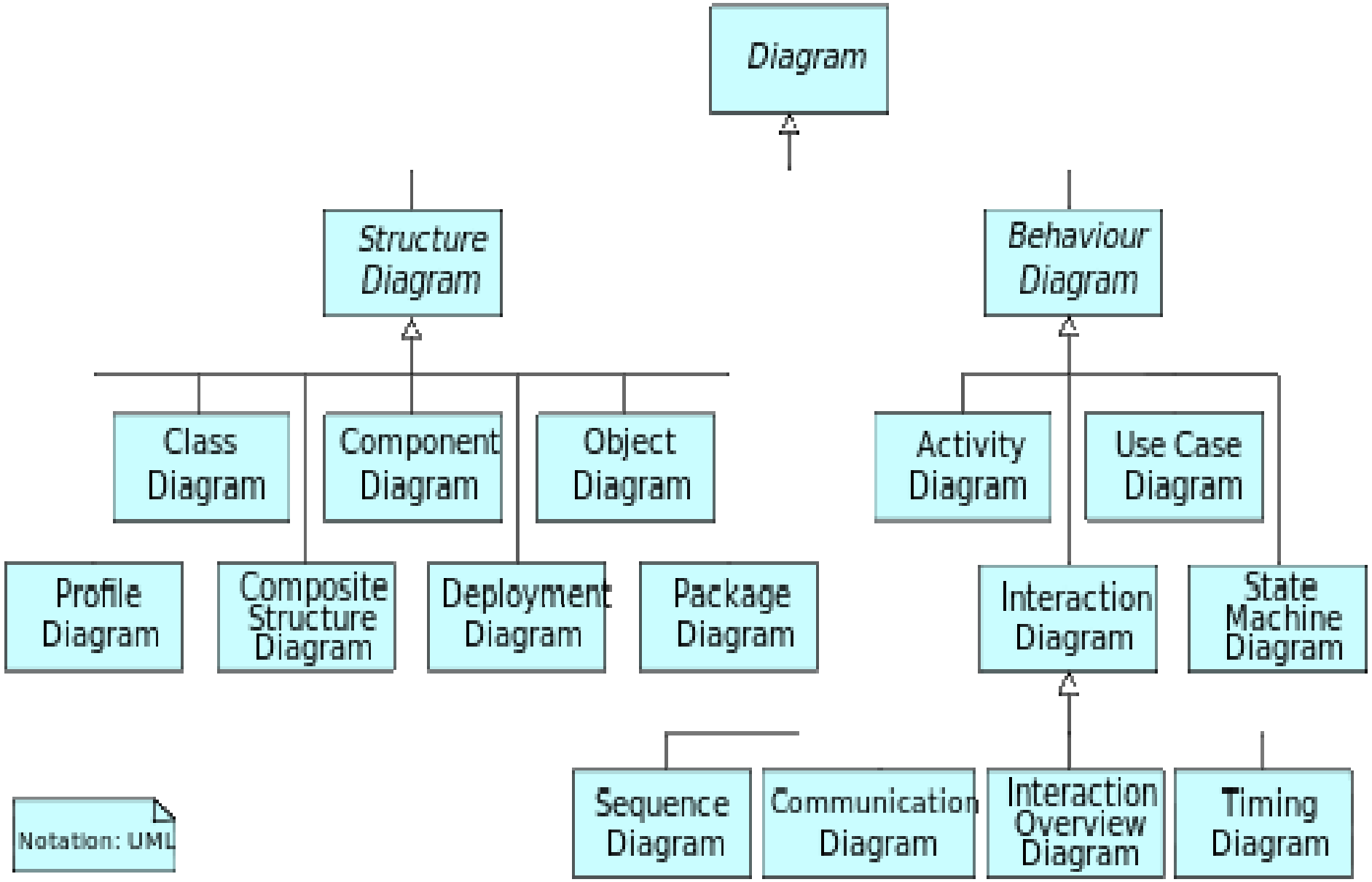
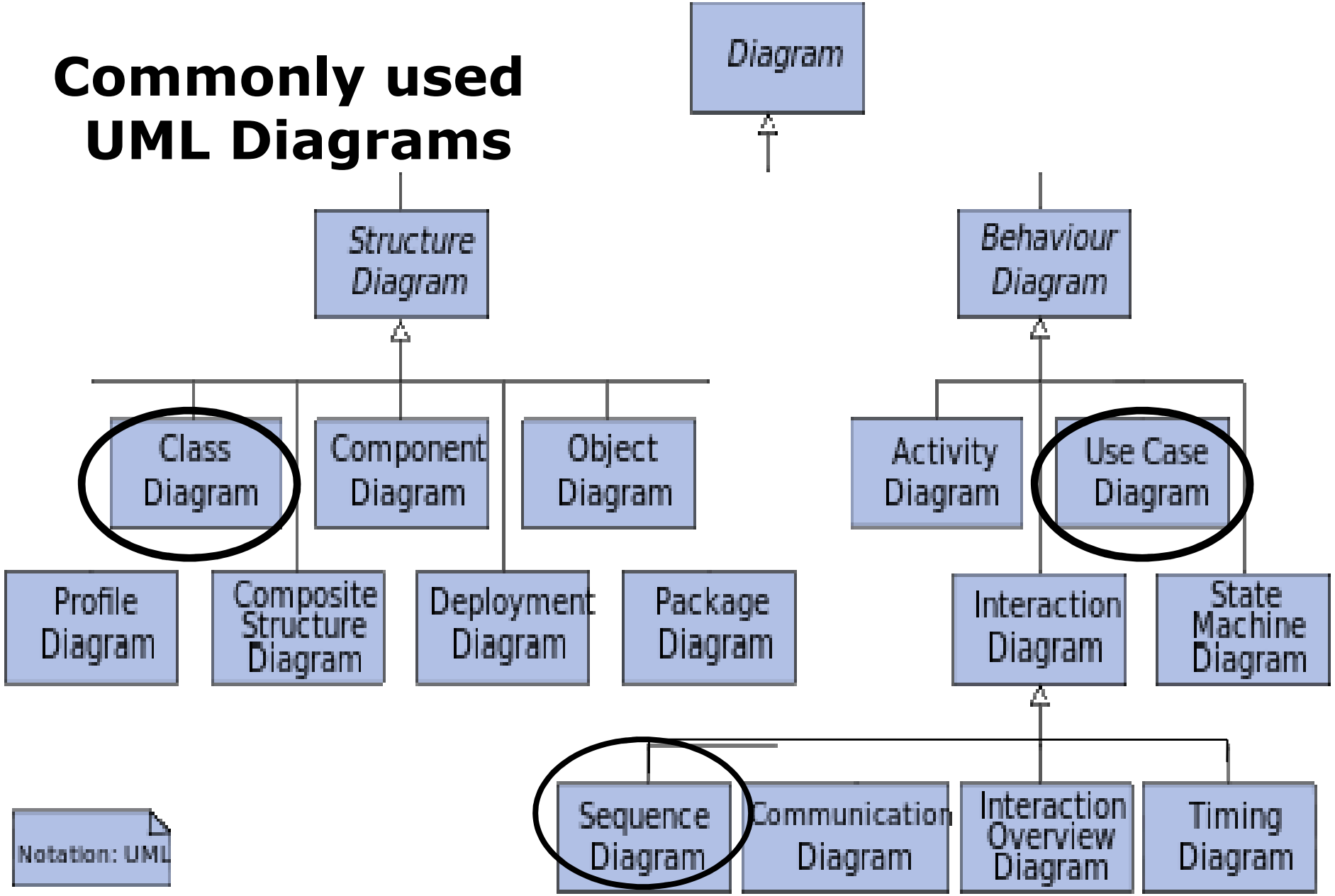


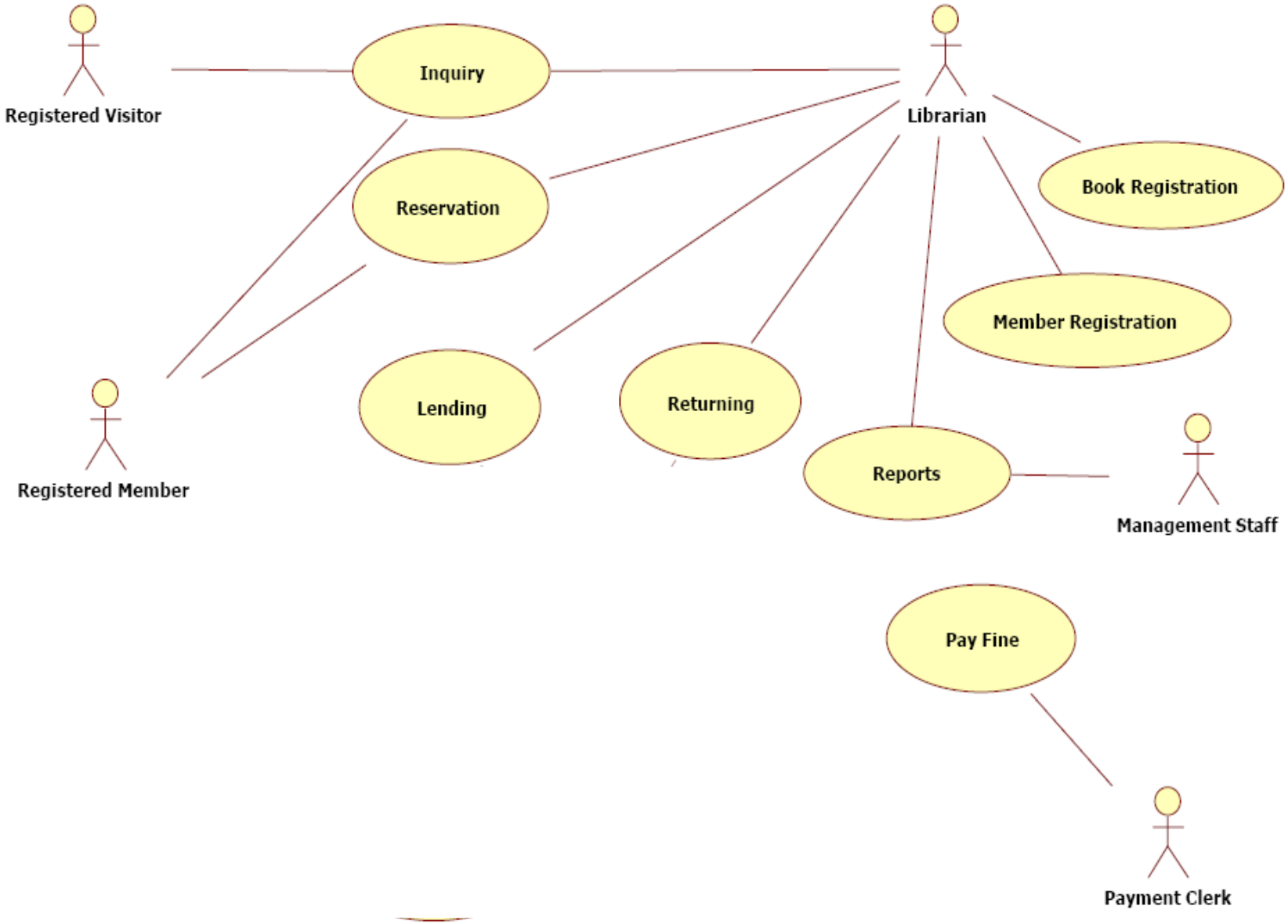
Diagram
Structure Diagram
Behaviour Diagram
Class Diagram
Component Diagram
Object Diagram
Activity Diagram
Use Case Diagram
Profile Diagram
Composite Structure Diagram
Deployment Diagram
Package Diagram
Interaction Diagram
State Machine Diagram
Sequence Diagram
Communication Diagram
Interaction Overview Diagram
Timing Diagram
Notation: UML

Commonly used UML Diagrams

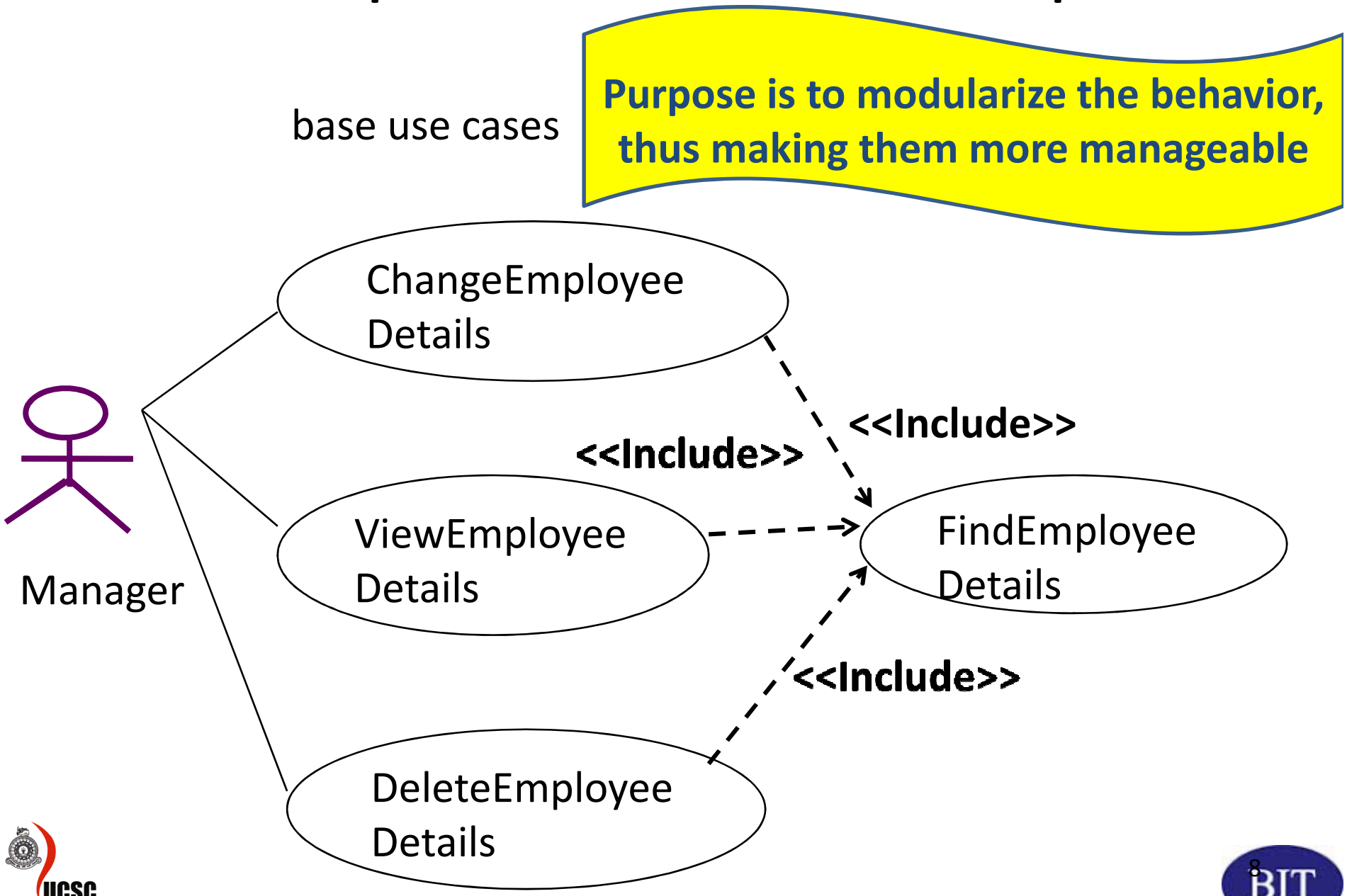


An Approach to OO Systems Analysis and Design.

- Model system requirements: Identify Actors and Use Cases and draw a top level Use Case diagram. (Analysis)
- You may also Identify Classes (Analysis)
- Document Use Case narratives (top level version initially) – Start During Analysis
- Draw Systems Sequence Diagram
- Start Designing Use Interfaces
- Complete Use Case Diagram, Document Use Case narratives (Expanded version), Draw Sequence Diagrams, Class Diagrams etc.


Registered Visitor
Inquiry
Librarian
Reservation
Book Registration
Member Registration
Registered Member
Lending
Returning
Reports
Management Staff
Pay Fine
Payment Clerk

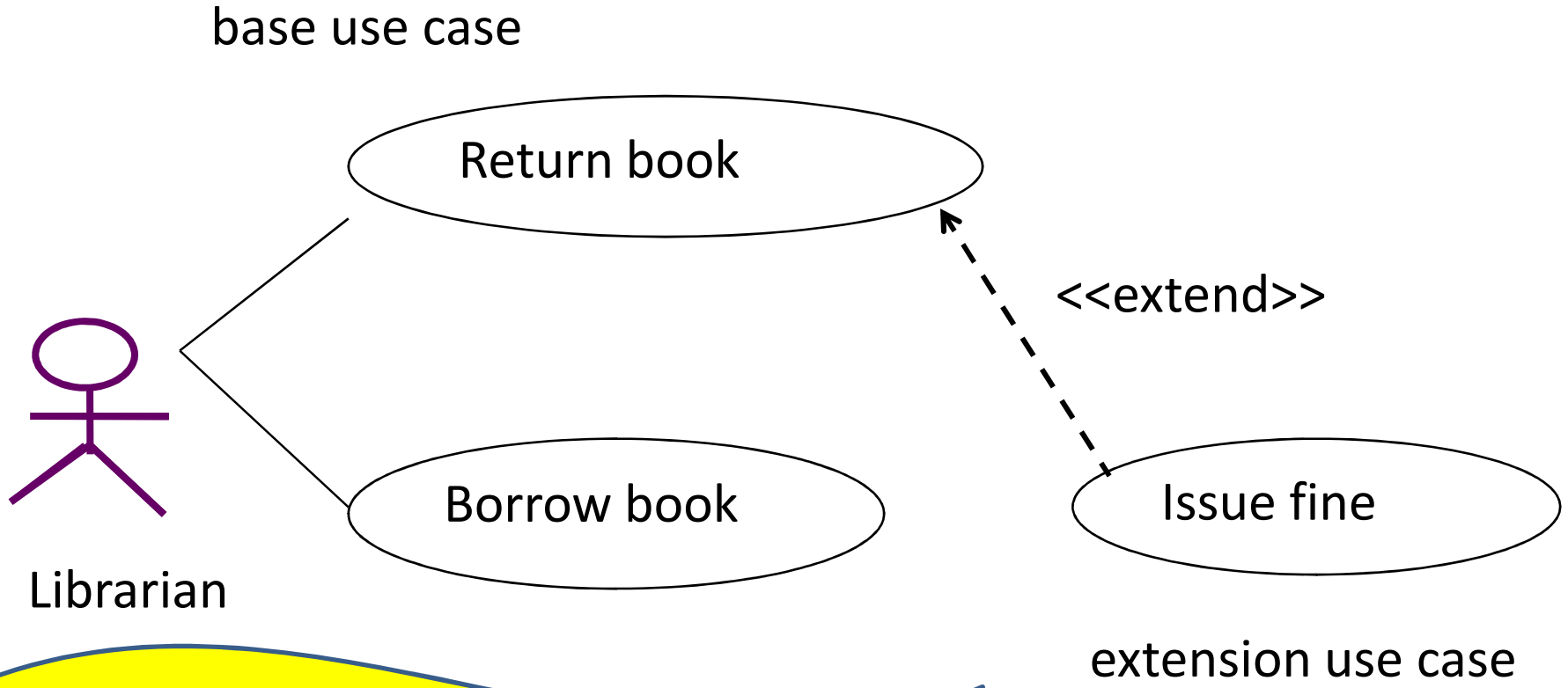
Example of <<include>> Relationship



Example of <<include>> Relationship cont..

- *Inclusion* use case supplies behavior to its base use case.
- The base use case executes until the point of inclusion is reached,
- Then execution passes over to the inclusion use case.
- When the inclusion use case finishes , the control return to the base use case again.

Example of <<extend>> Relationship

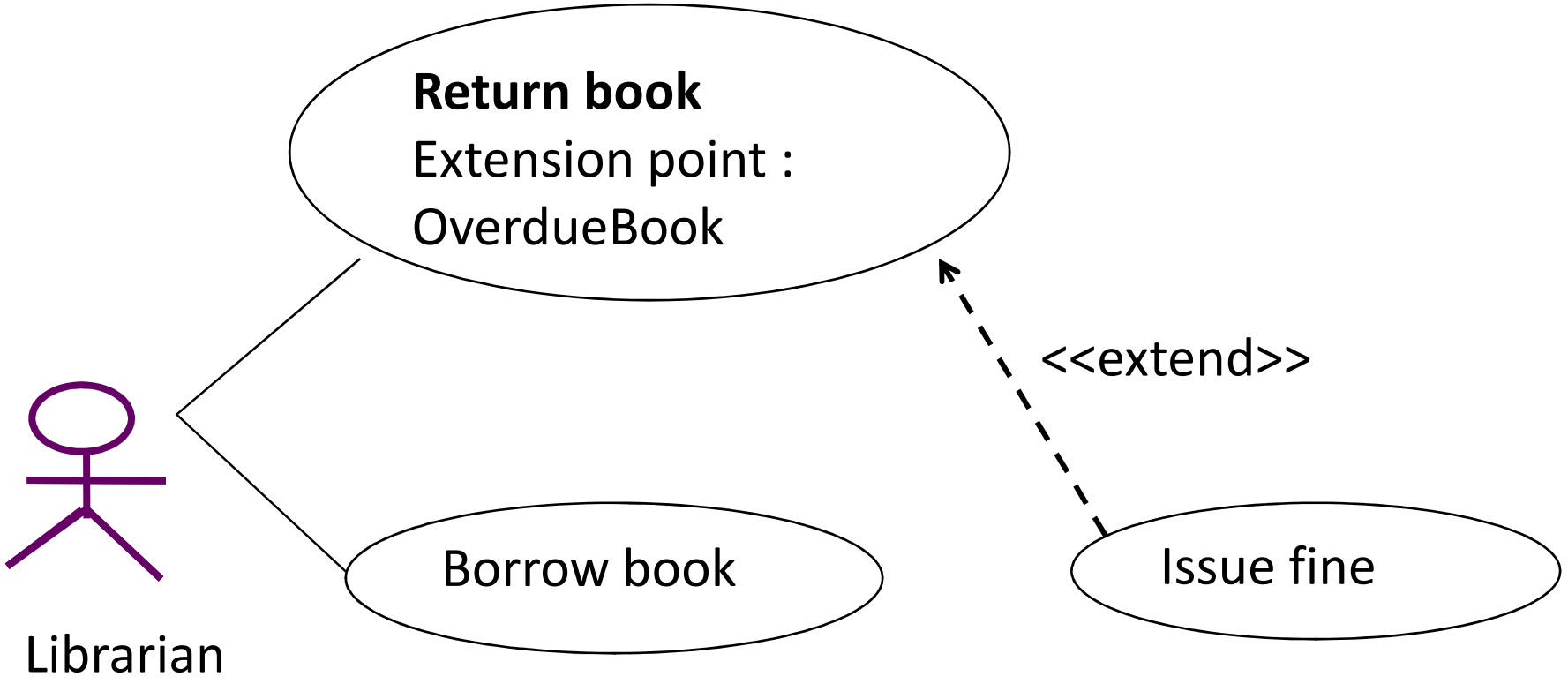


Provides a way to insert new behavior into an existing use case

Example of <<extend>> Relationship cont..

- <<extend>> provides a way to insert new behaviour into an existing use case.
- The base use case provides a set of extension points that are hooks where new behaviour is added.
- Extension use case provides a set of insertion segments that can be inserted into the base use case at these hooks.

Example of <<extend>> Relationship cont..



Documenting Use Case Narratives (High Level)

Use-Case Name:		
Use-Case ID:		**Use-Case Type**
Priority:		**Business Requirements:** ☐
Source:		
Primary Business Actor:		
Other Participating Actors:		
Other Interested Stakeholders:		
Description:		

Importance of the Use Case – typically high , medium , low

Documenting Use Case Narratives (High Level)

Use-Case Name:		**Use-Case Type**
Use-Case ID:		**Business Requirements:**
Priority:		
Source:		☐
Primary Business A		
Other Participating Actors:		
Other Interested Stakeholders:		
Description:		

Entity that triggers the creation of the Use Case. Eg. Document

BIT

Documenting Use Case Narratives (High Level)

Use-Case Name:		**Use-Case Type**
Use-Case ID:		**Business Requirements:**
Priority:		
Source:		
Primary Business Actor:		
Other Participating Actors:		
Other Interested Stakeholders:		
Description:		

Who benefits from the use case

BIT

Documenting Use Case Narratives (High Level)

Use-Case Name:		**Use-Case Type**
Use-Case ID:		**Business Requirements:**
Priority:		
Source:		
Primary Business Actor:		
Other Participating Actors:		
Other Interested Stakeholders:		
Description:		

Facilitating Actors

BIT

Documenting Use Case Narratives (High Level)

Use-Case Name:		**Use-Case Type** **Business Requirements:** ☐
Use-Case ID:		
Priority:		
Source:		
Primary Business Actor:		
Other Participating Actors:		
Other Interested Stakeholders:		
Description:		

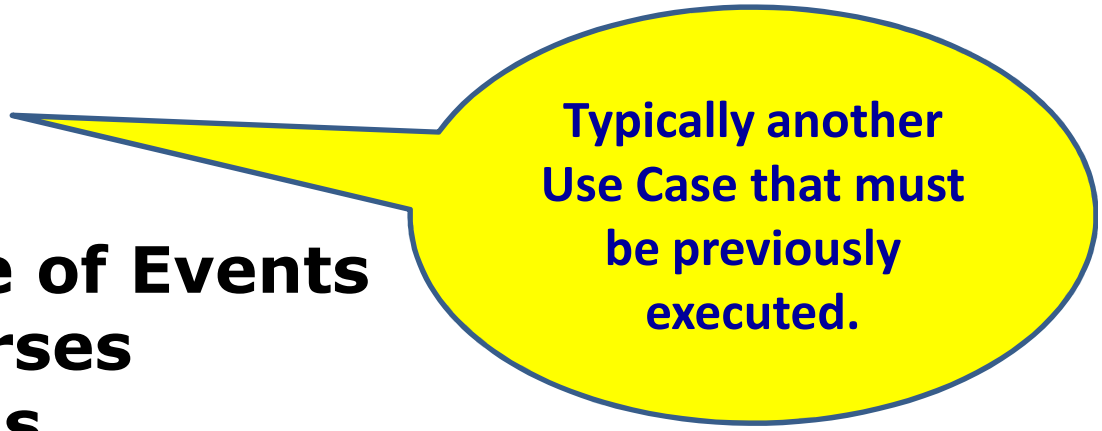
General understanding of problem domain and scope

In brief

BIT

Documenting Use Case Narratives (Expanded version)

- **Preconditions**
- **Trigger**
- **Typical Course of Events**
- **Alternate Courses**
- **Post conditions**

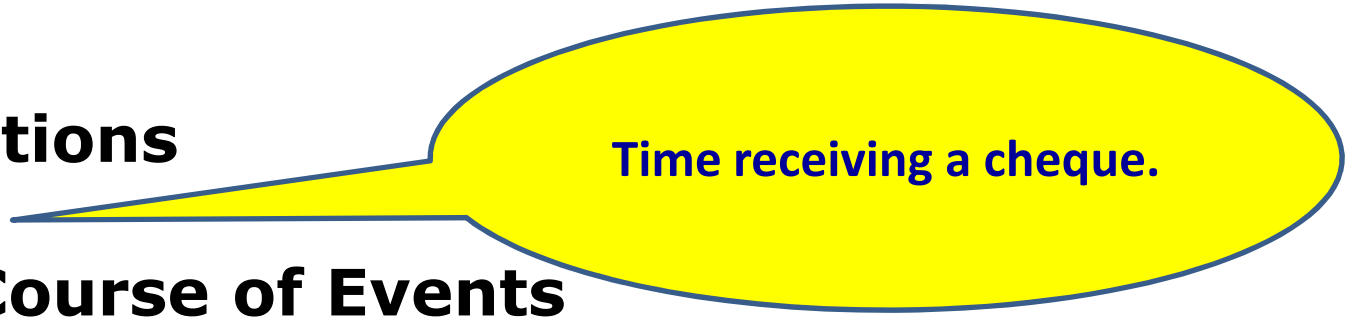


BIT

Documenting Use Case Narratives (Expanded version)

- **Preconditions**
- **Trigger**
- **Typical Course of Events**
- **Alternate Courses**
- **Post conditions**

etc. are included.

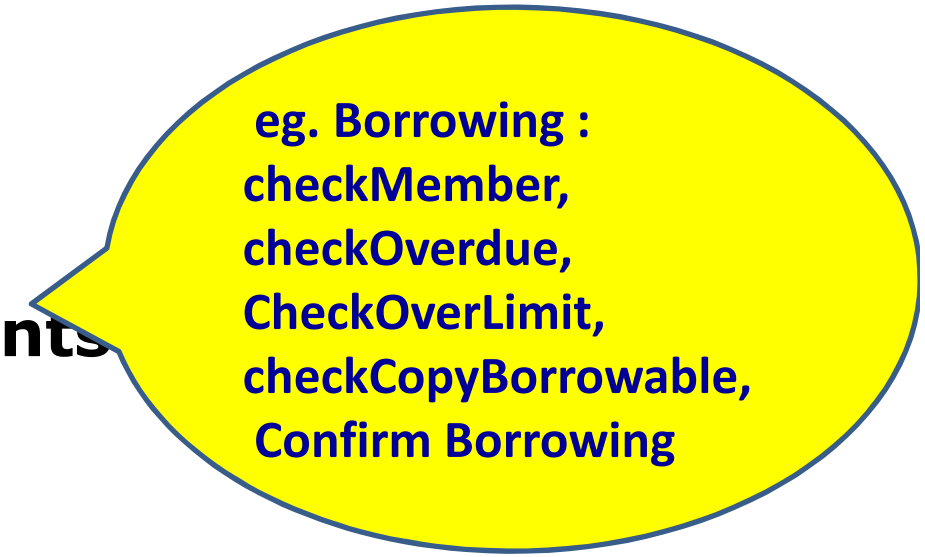


BIT

Documenting Use Case Narratives (Expanded version)

- **Preconditions**
- **Trigger**
- **Typical Course of Events**
- **Alternate Courses**
- **Post conditions**

etc. are included.



BIT

Documenting Use Case Narratives (Expanded version)

- **Preconditions**
- **Trigger**
- **Typical Course of Events**
- **Alternate Courses**
- **Post conditions**

etc. are included.

BIT

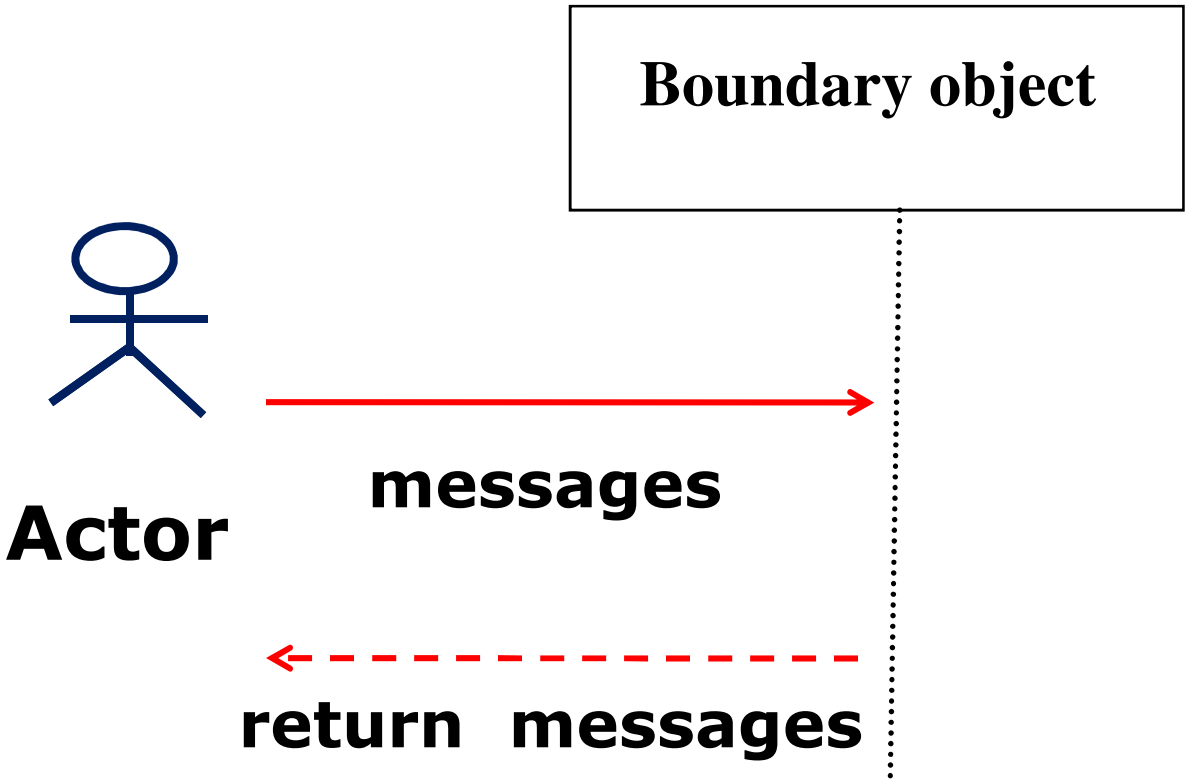
Documenting Use Case Narratives (Expanded version)

- **Preconditions**
- **Trigger**
- **Typical Course of Events**
- **Alternate Courses**
- **Post conditions**

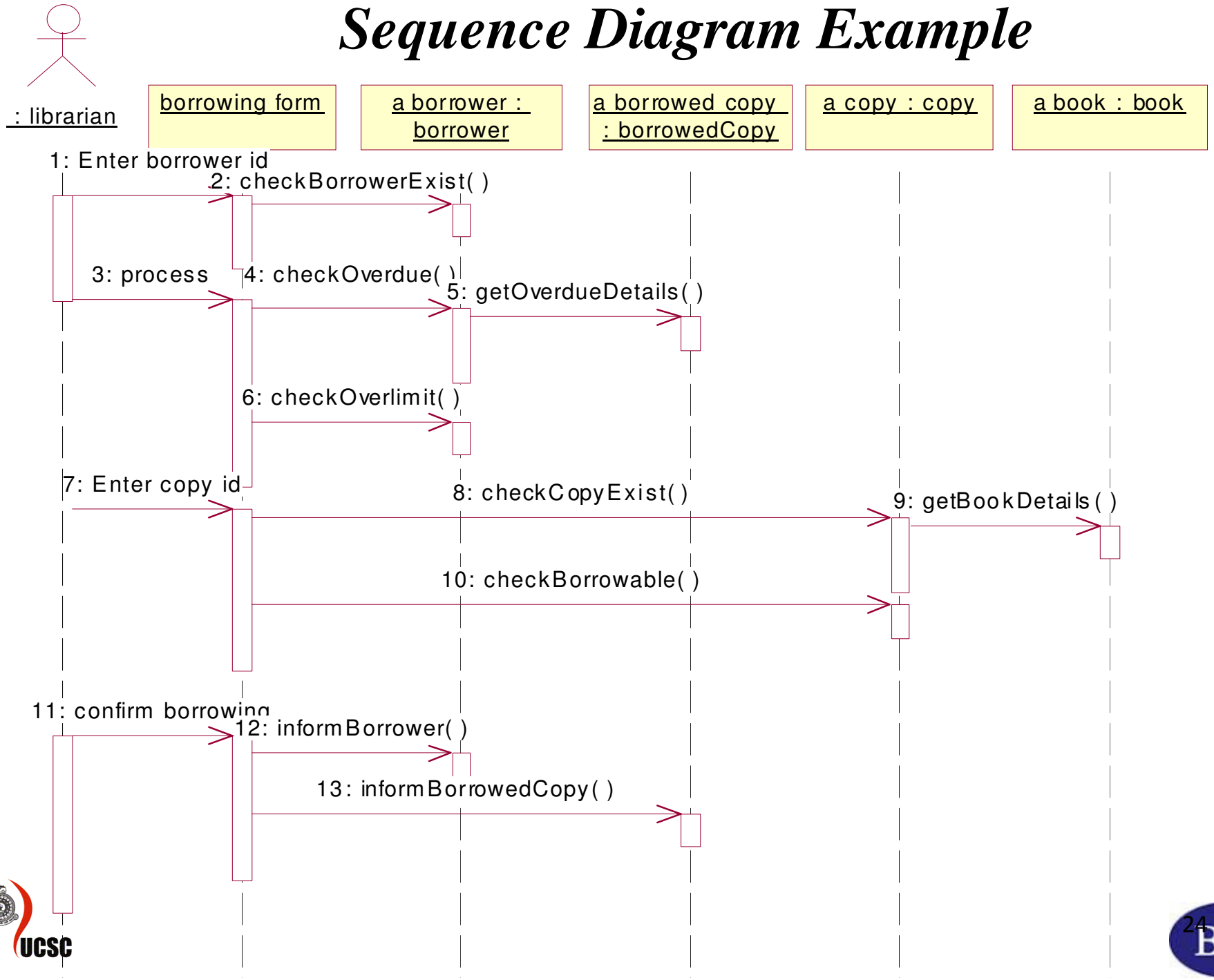
etc. are included.

BIT

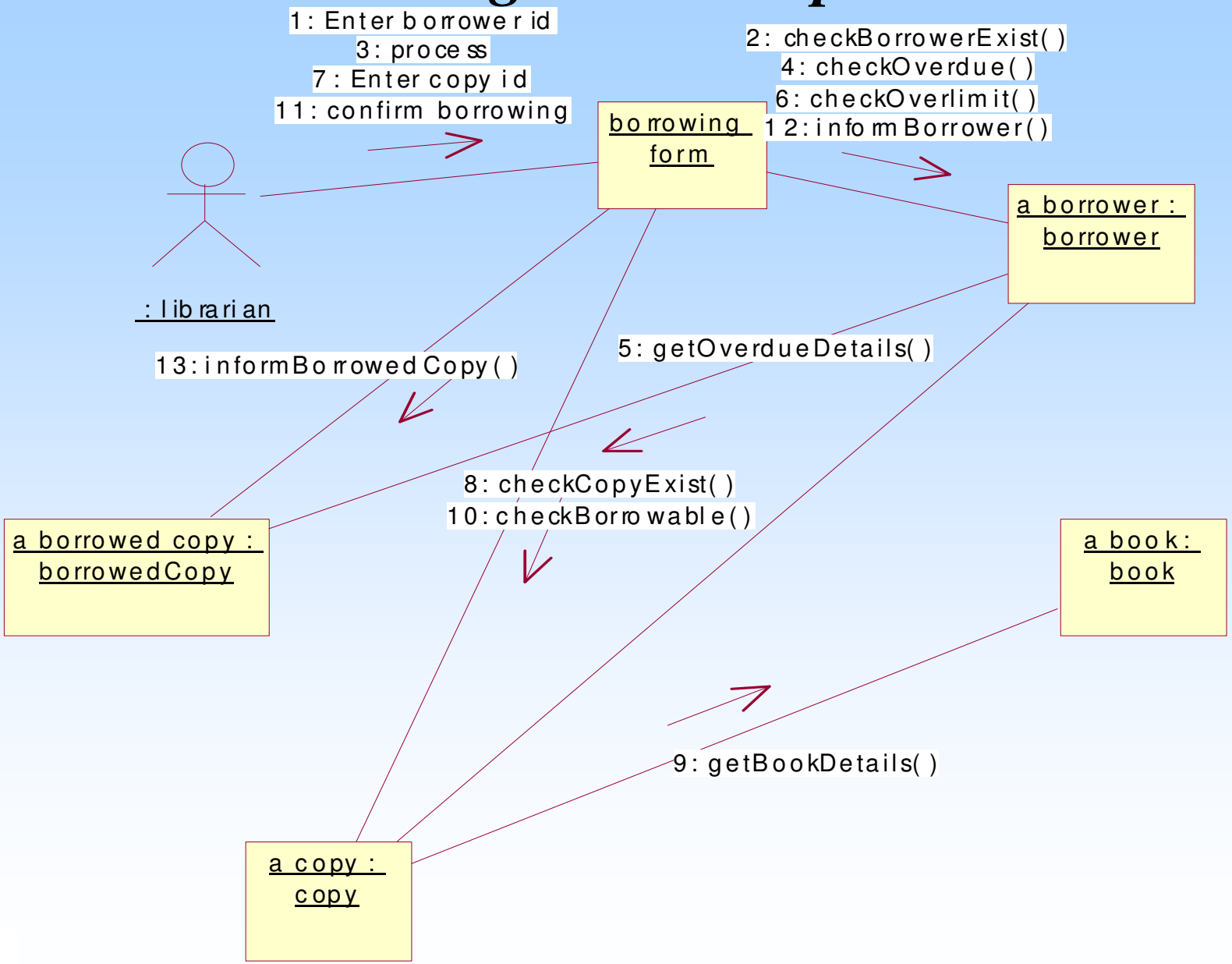
System Sequence Diagrams

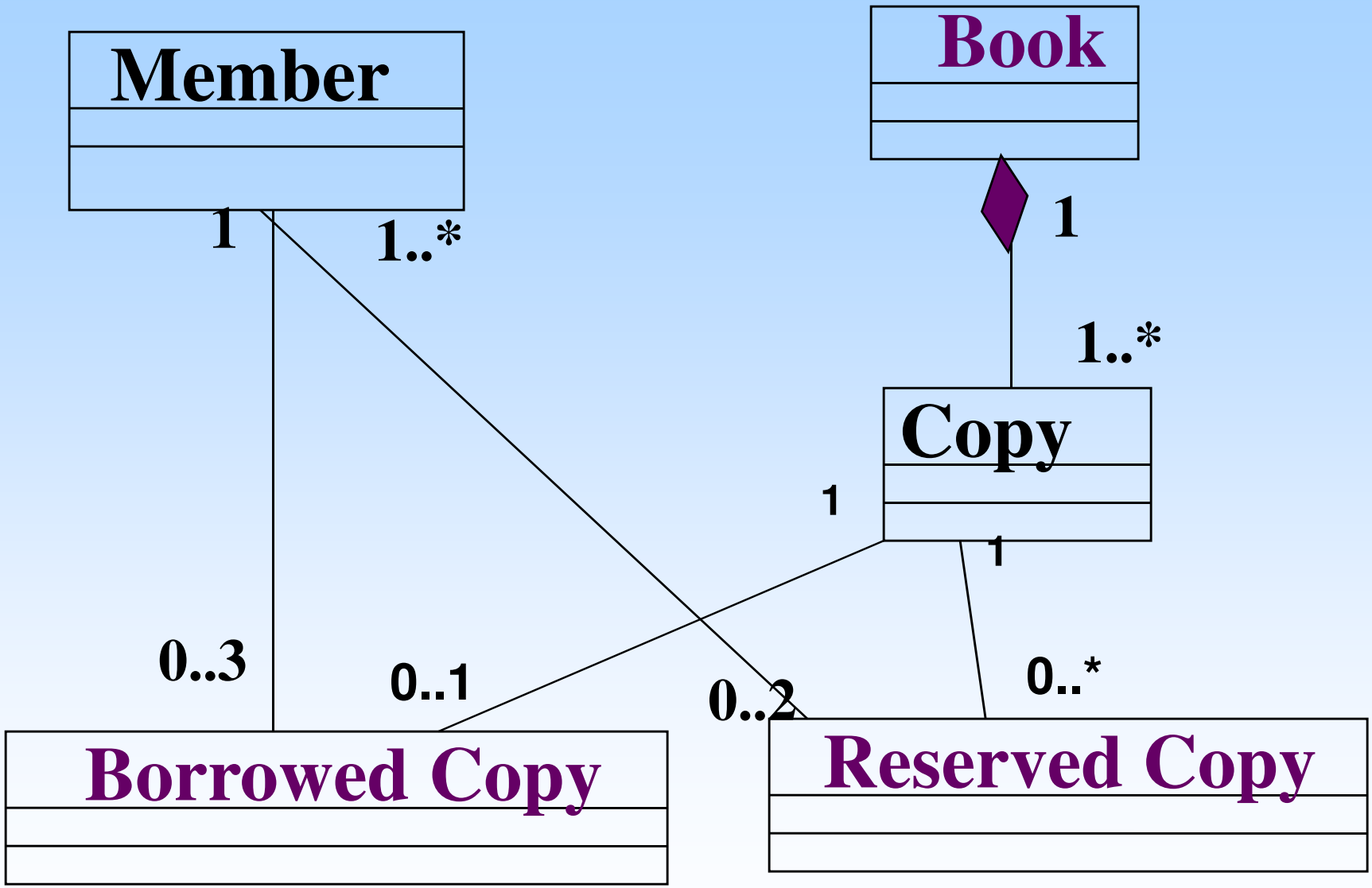


Sequence Diagram Example



Collaboration Diagram Example





Class Diagram for a library system

Questions (T/F):Use Case Diagrams

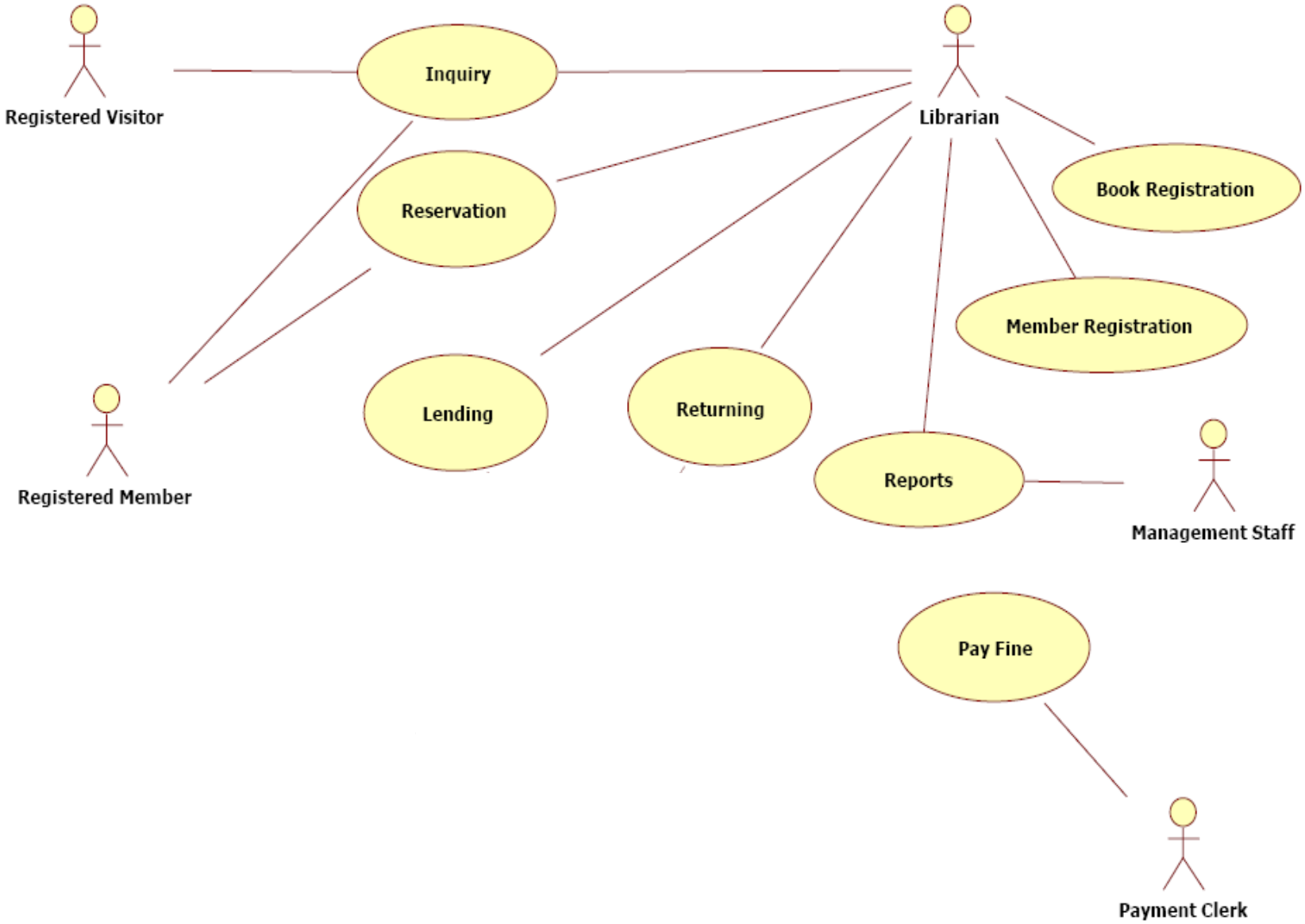
(i) The collection of Use Cases for a system constitute all the defined ways in which the system may be used.

(ii) Time can be considered as an actor in a Use Case model.

(iii) A Use Case describes what a system does and how it is done.

(iv) Use Case diagrams provide a simple and easily understood way for clients to view their requirements.

(v) Actors in a Use Case model represent anyone or anything that must interact with the system.


Registered Visitor
Inquiry
Librarian
Reservation
Book Registration
Member Registration
Registered Member
Lending
Returning
Reports
Management Staff
Pay Fine
Payment Clerk

Questions (T/F):Use Case Diagrams

(i) The collection of Use Cases for a system constitute all the defined ways in which the system may be used. ✓

(ii) Time can be considered as an actor in a Use Case model. ✓

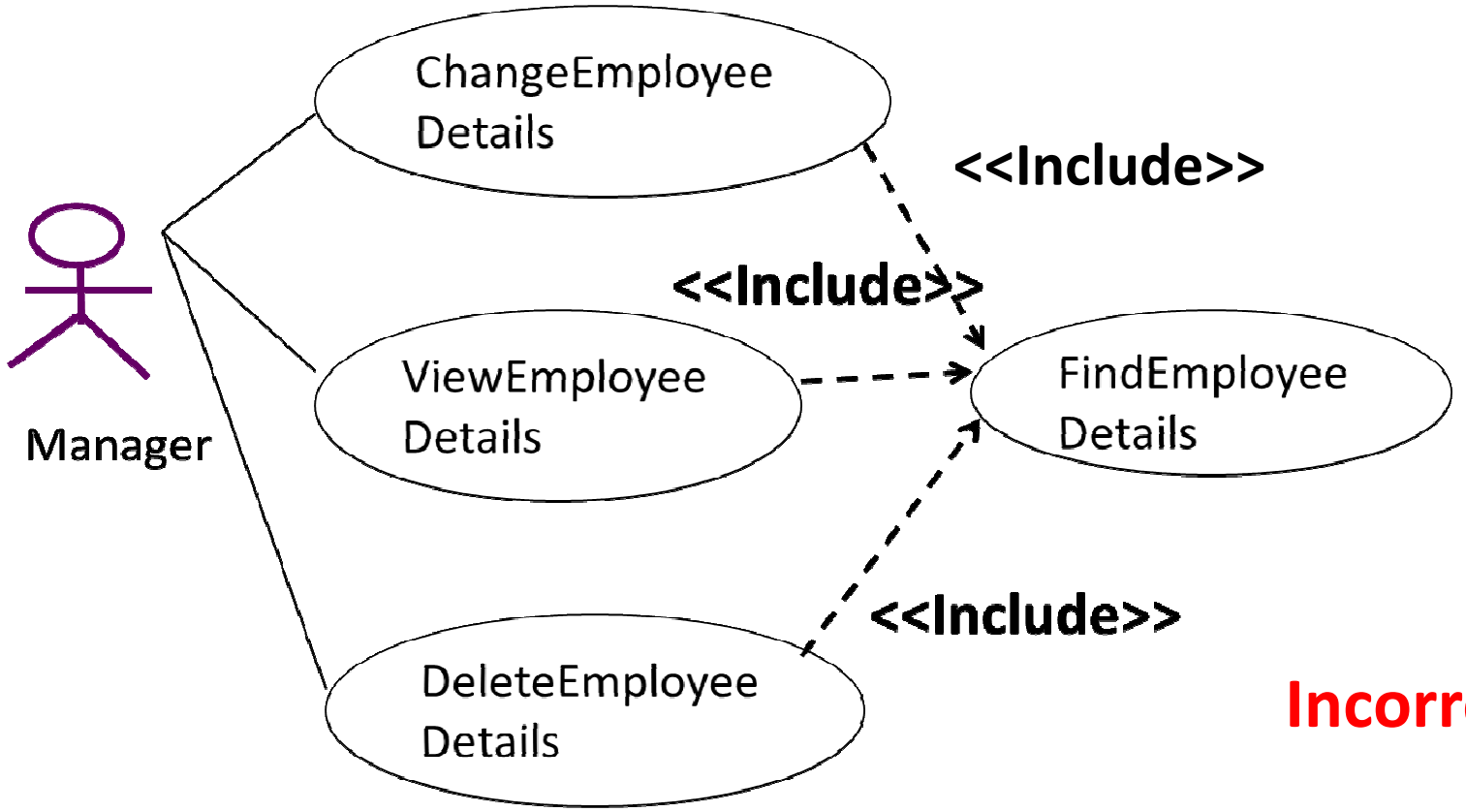
(iii) A Use Case describes what a system does and how it is done. ✗

(iv) Use Case diagrams provide a simple and easily understood way for clients to view their requirements. ✓

(v) Actors in a Use Case model represent anyone or anything that must interact with the system. ✓

Questions (T/F):Use Case Diagrams cont..

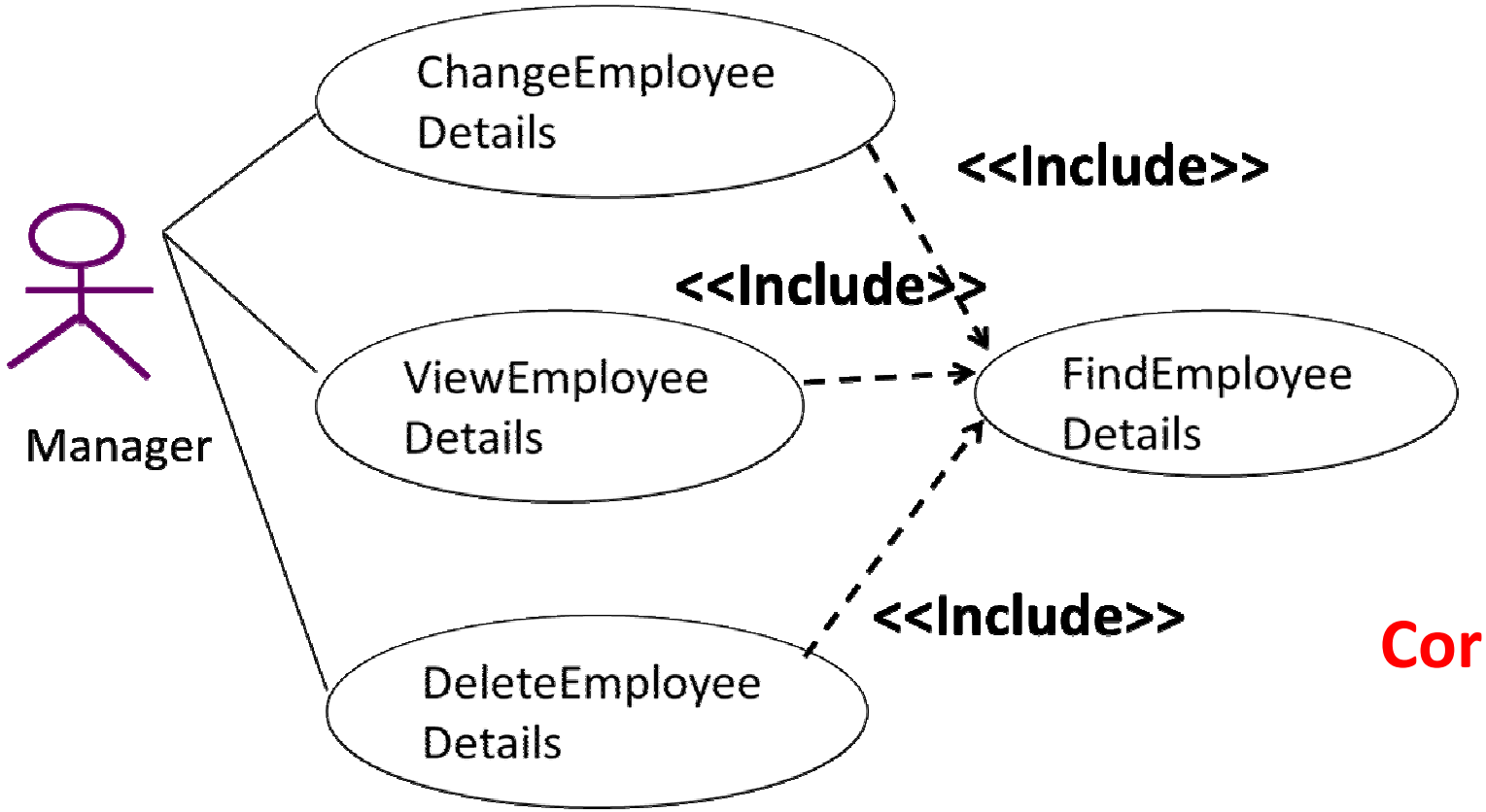
(vi) Inclusion use case supplies behavior to its base use case optionally.



Incorrect

Questions (T/F):Use Case Diagrams cont..

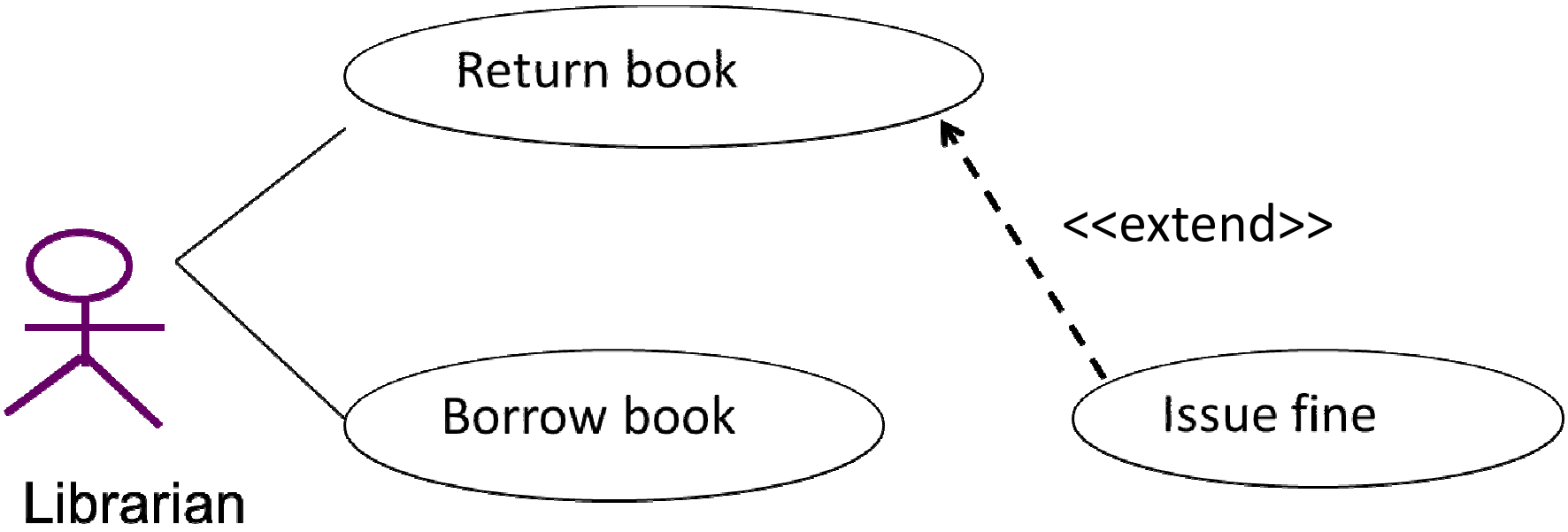
(vii) When the inclusion use case in a use case diagram finishes, the control return to the base use case.



Correct

Questions (T/F):Use Case Diagrams cont..

(viii) <<extend>> provides a way to insert new behavior into an existing use case.



Correct

Identifying Classes and drawing Class Diagrams.

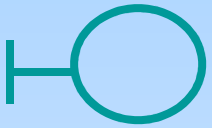
- There are three primary class stereotypes in UML.

Boundary

Entity

Control

Stereotypes and Classes cont…

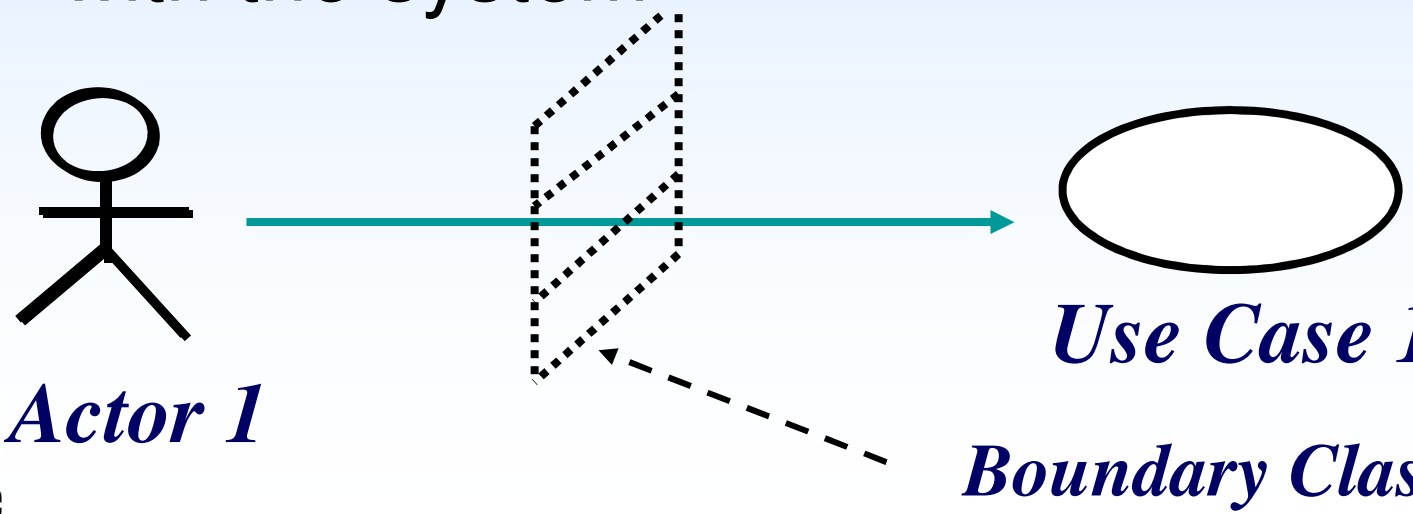
***Boundary** Class:* 

- They provide the interface to a user or another system. (ie. Interface to an actor).
- Handles communication between system surroundings and the inside of the system.
- To find the *Boundary* classes, you can examine your Use Case diagram,

Stereotypes and Classes cont…

Boundary Class:

- At a minimum there must be, one *Boundary* class for every actor-use case interaction.
- Boundary class allows actor to interact with the system.



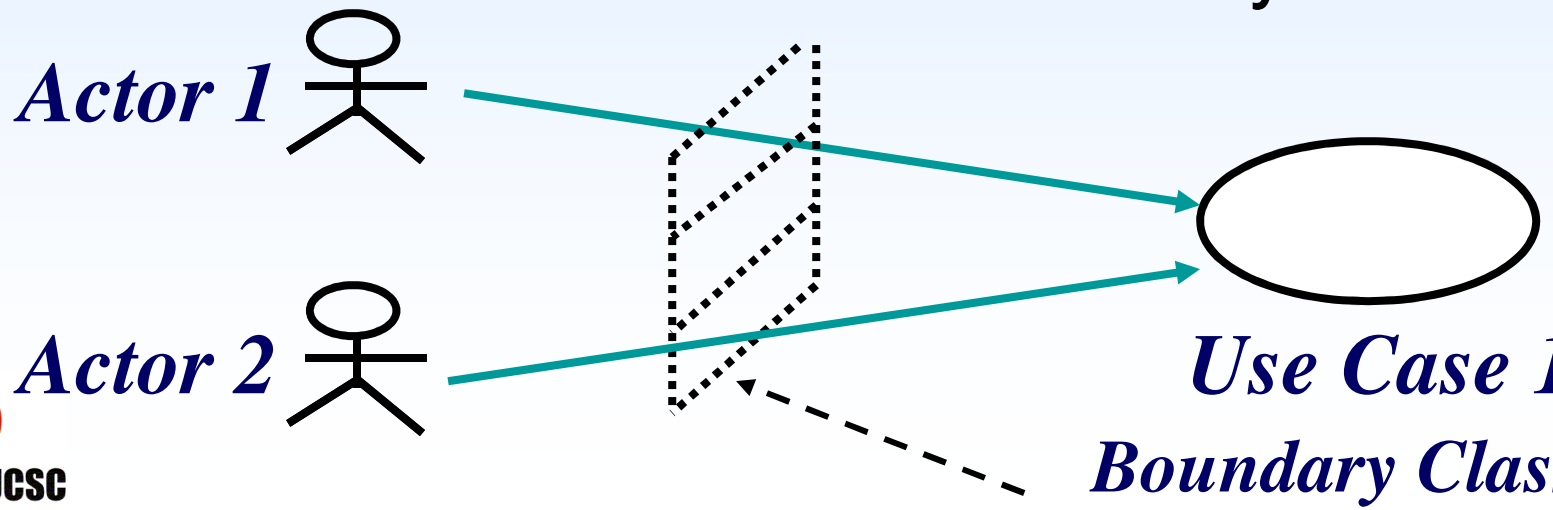
UCSC

BIT

Stereotypes and Classes cont…

Boundary Class:

- You do not necessarily have to create a unique *Boundary* class for every actor-use case pair.
- Two actors may initiate the same use case.
- They might both use the same Boundary class to communicate with the system.



Finding Boundary Classes

- These are classes that mediate between the subject (System boundary) and its environment.
  - User Interface class – classes that interface between the system and humans;
  - System Interface class – classes that interface with other systems;
  - Device Interface class – classes that interface with external devices such as sensors;

Stereotypes and Classes cont..

Entity Class 

- They are needed to perform task internal to the system. Reflect a real world entity.

Identifying Entity Classes

Noun/Verb Analysis

- Identify the nouns and noun phrases

Stereotypes and Classes cont..

Entity Class

- The initial list of nouns must be filtered because,
  - it could contain nouns that are outside the problem domain.
  - nouns that are just language expressions.
  - nouns that are redundant.
  - nouns that are attributes.

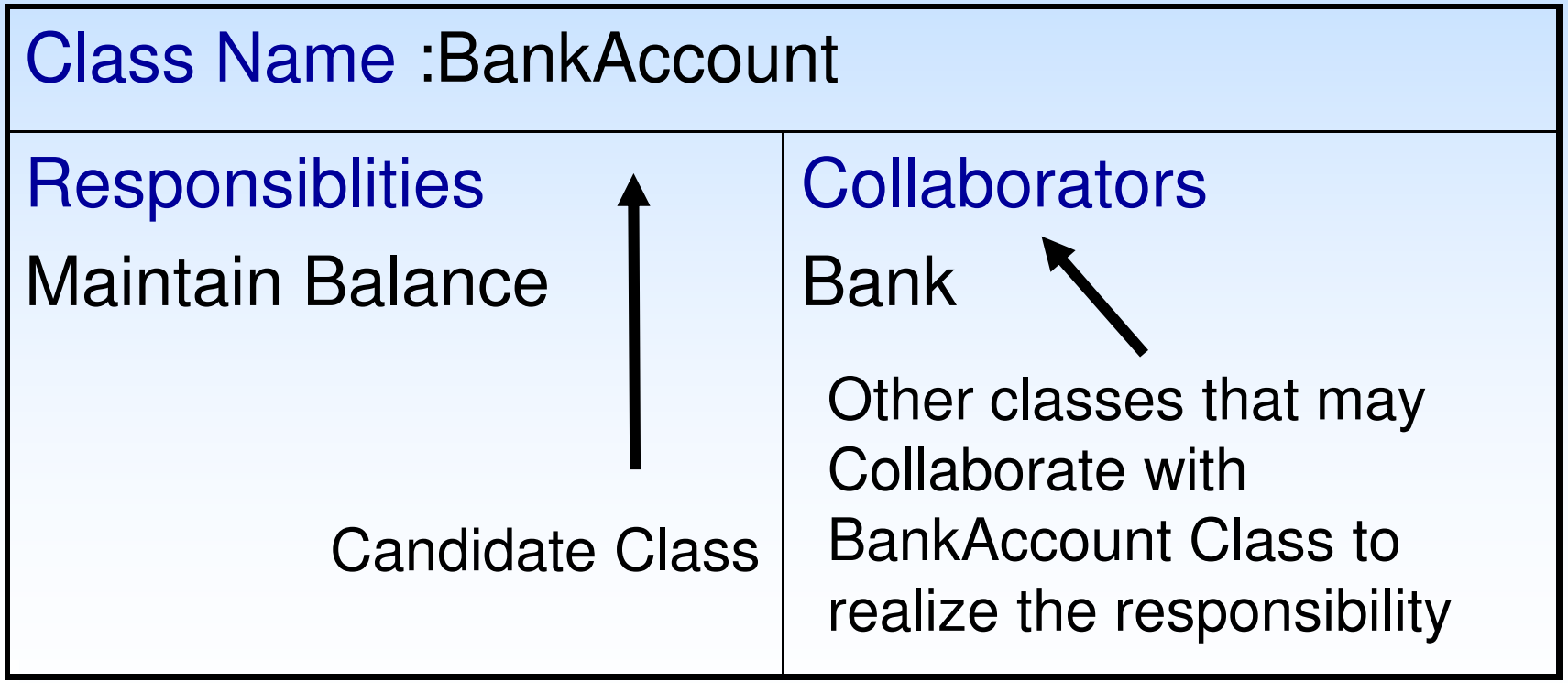
Stereotypes and Classes cont..

Entity Class

Using CRC Analysis

- CRC – Class, Responsibilities and Collaborators

eg



Stereotypes and Classes cont..

Control Class:

- Sequencing behaviour specific to one or more use cases.
- There is typically one *control* class per use case.
- Co-ordinates the events needed to realise the behaviour specified in the use case.

  Eg. Running or *executing* the use case.

Questions (T/F): Class Diagrams

(i) Association relationship in a class diagram should always show the navigability. ✗

(ii) Composition relationship is drawn as a filled diamond. ✓

(iii) Association names should be noun phrases because they indicate an action that the source object is performing on the target object. ✗

(iv) Multiplicity specifies the number of objects that can participate in a relationship at any point of time.? ✓

UCSC BIT

Multiplicity Indicators

1 Exactly one

0..* Zero or more

1..* One or more

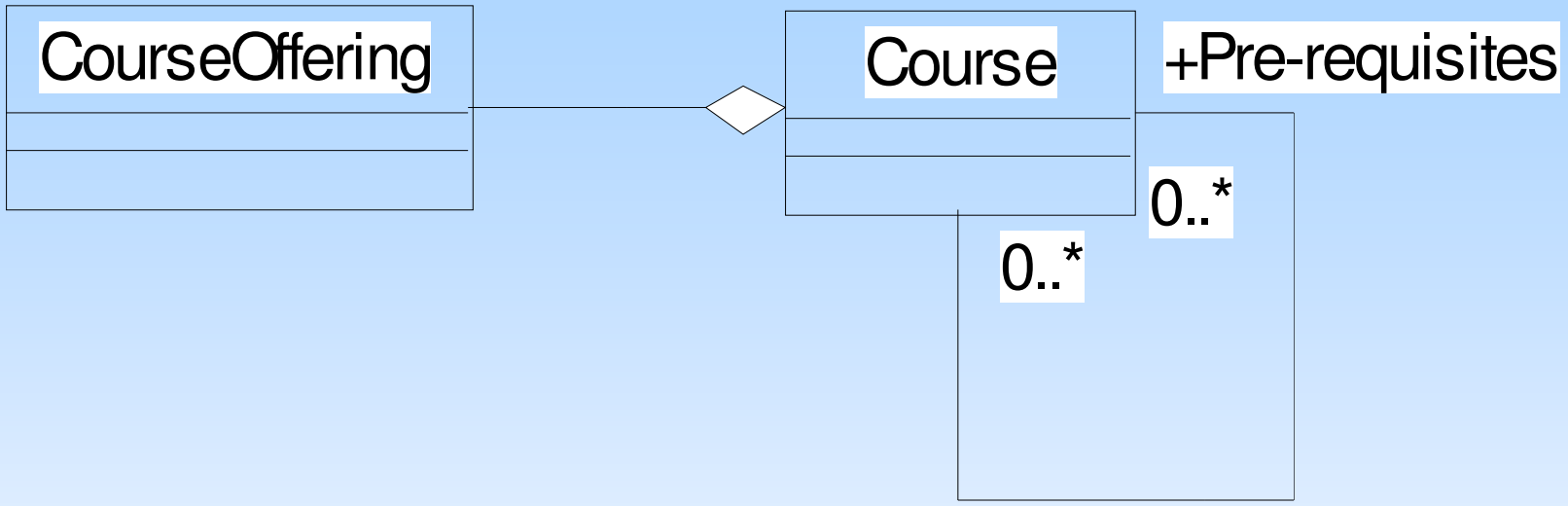
0..1 Zero or one

5..8 Specific Range (5,6,7 or 8)

Reflexive relationships

- Some times a class is in an association with itself.
- This can happen when a class has objects that can play a variety of roles.
- This is shown on the class diagram as a reflexive association or aggregation.
- Role names rather than association names are typically used for reflexive relationships.

Reflexive relationships cont..



- One Course object playing the role of Prerequisite is related to zero or more course objects.
- One Course object is related to zero or more course objects playing the role of Prerequisite.

Association Classes

- A relationships between objects may also have structure and behaviour.
- This happens for links between two objects, not with one object by itself.
- When this happens, UML provides a facility to held the structure and behaviour belonging to the relationship, in an **association class**.

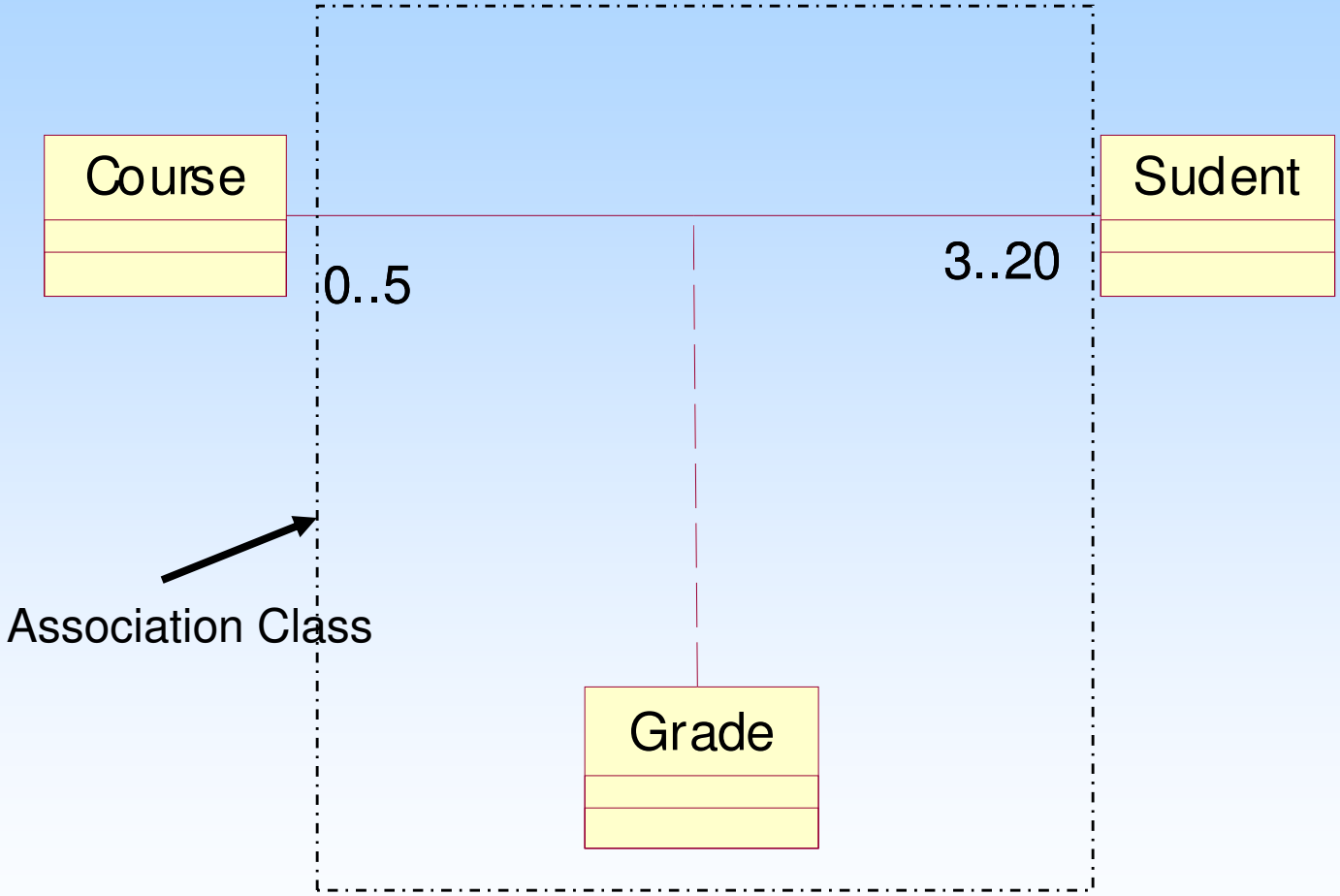
Association Class cont….

A student may take up to 5 courses.
A course may have between 3 and 20 students.
Each student must receive a grade for the course
Where is the grade held?

It belongs to the link.

Association Class

eg.



Which of the above would form super class-subclass pairs?

(i) Employee, WeeklyPaid Employee ✓

(ii) Savings Account, Current Account

(iii) Money, Singapore Dollars ✓

(iv) Student, Post Graduate Student ✓

(v) Account, Account324567

(vi) Region, City

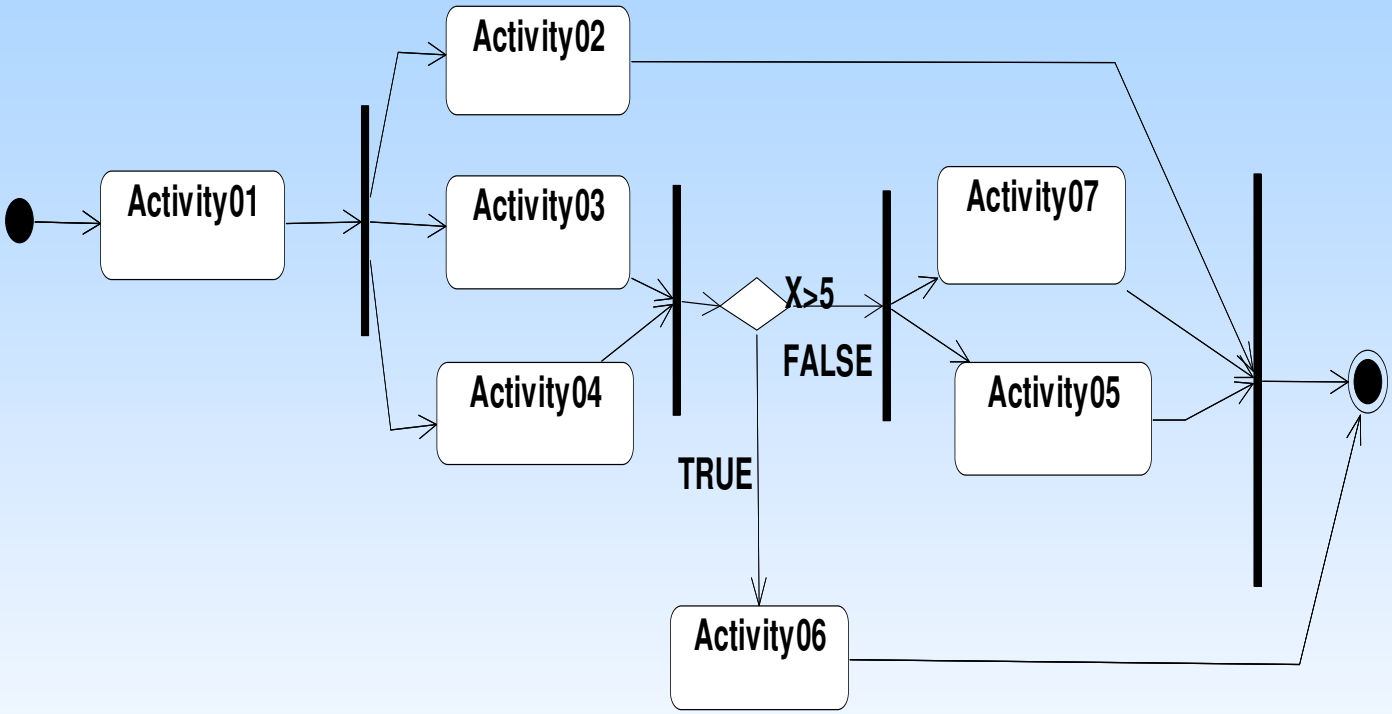
(vii) Account, Fixed Deposit Account ✓

(viii)Aircraft , Engine

(ix) Book, Chapter

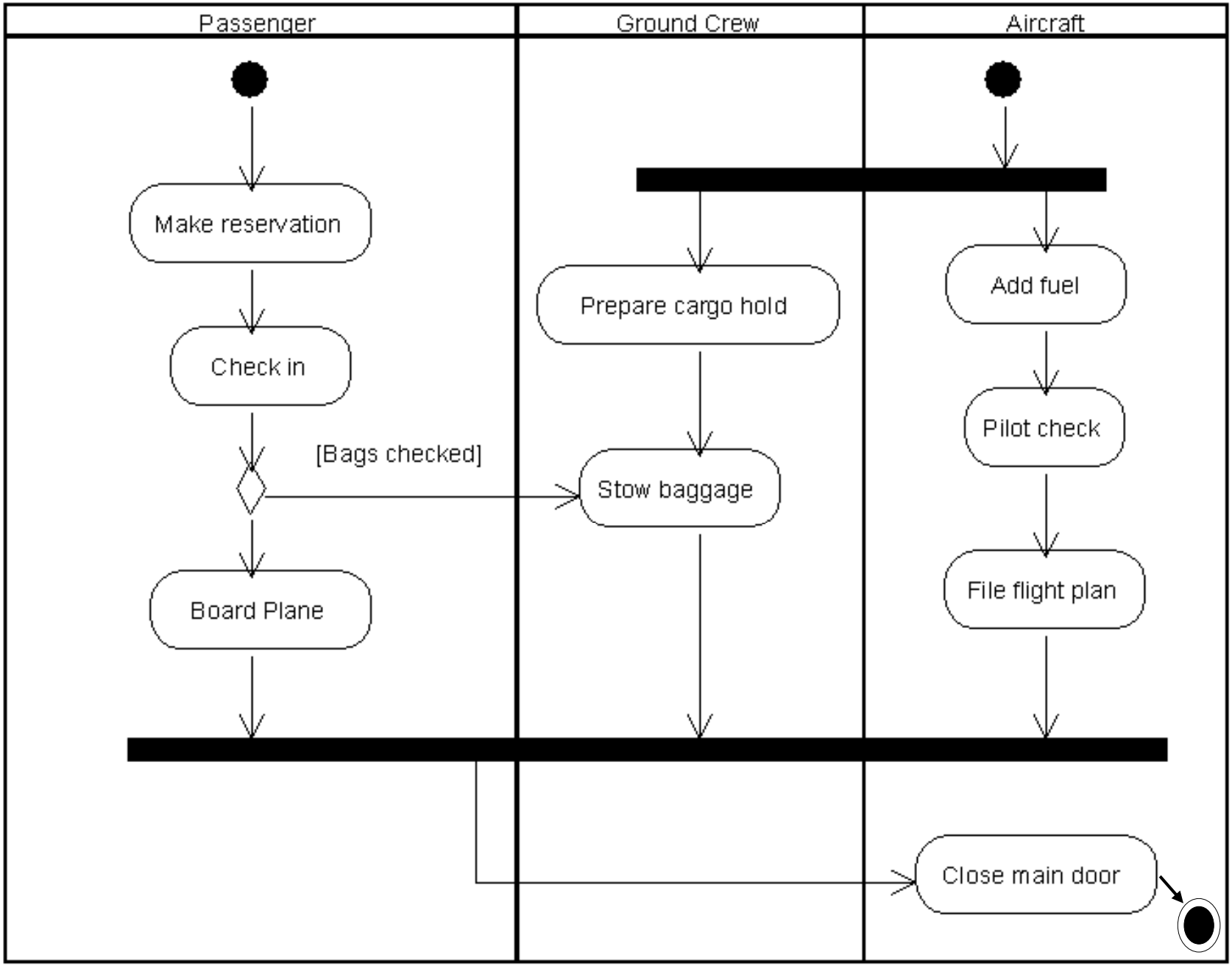
(xiv)Payment, corporate billing ✓

Activity Diagrams



- Identify activities that happen in parallel.

3,4 and 7,5,2


Passenger
Ground Crew
Aircraft
Make reservation
Check in
[Bags checked]
Board Plane
Prepare cargo hold
Stow baggage
Add fuel
Pilot check
File flight plan
Close main door

Working with State Diagrams

- UML *State Transition Diagrams* shows:
  - Life history showing the different states of a given object.
  - The events or messages that cause a transition from one state to another.
  - The actions that results from a state change.
- *State Diagrams* are created only for classes with significant dynamic behaviour.

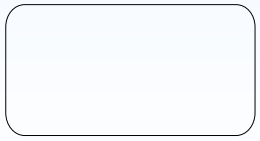
  eg. ***Hotel Room*** in a Hotel Reservation System

Modeling Dynamic Behaviour

- Interaction diagrams can be studied to determine the dynamic objects.
  - Objects receiving and sending many messages.
- If you have an attribute called *status*.
  - This can be a good indicator of various states.

States

- eg. HotelRoom object can be in one of the following states.
  - Occupied, Available, Reserved
- eg. Course object (in a course registration system) can be in one of the following states.
  - Initialization,Open, Close,Cancel



UML Notation for a State

State Transitions

- A State Transition represents a change from an originating state to a successor state.
- An action can accompany a state transition.
- A State Transition is represented by an arrow that points from the originating state to the successor state.

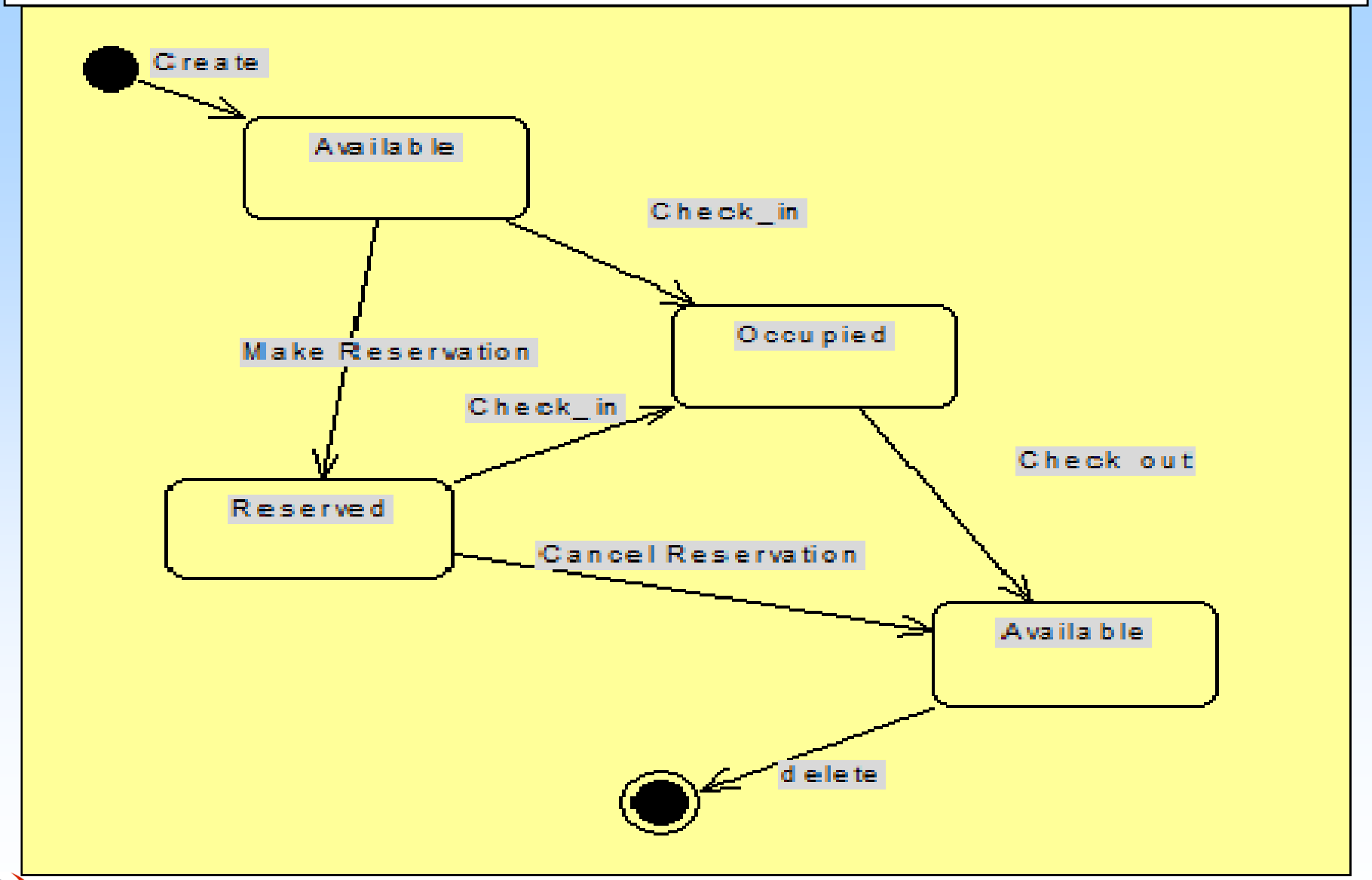
UML Notation for State Transition

Special States

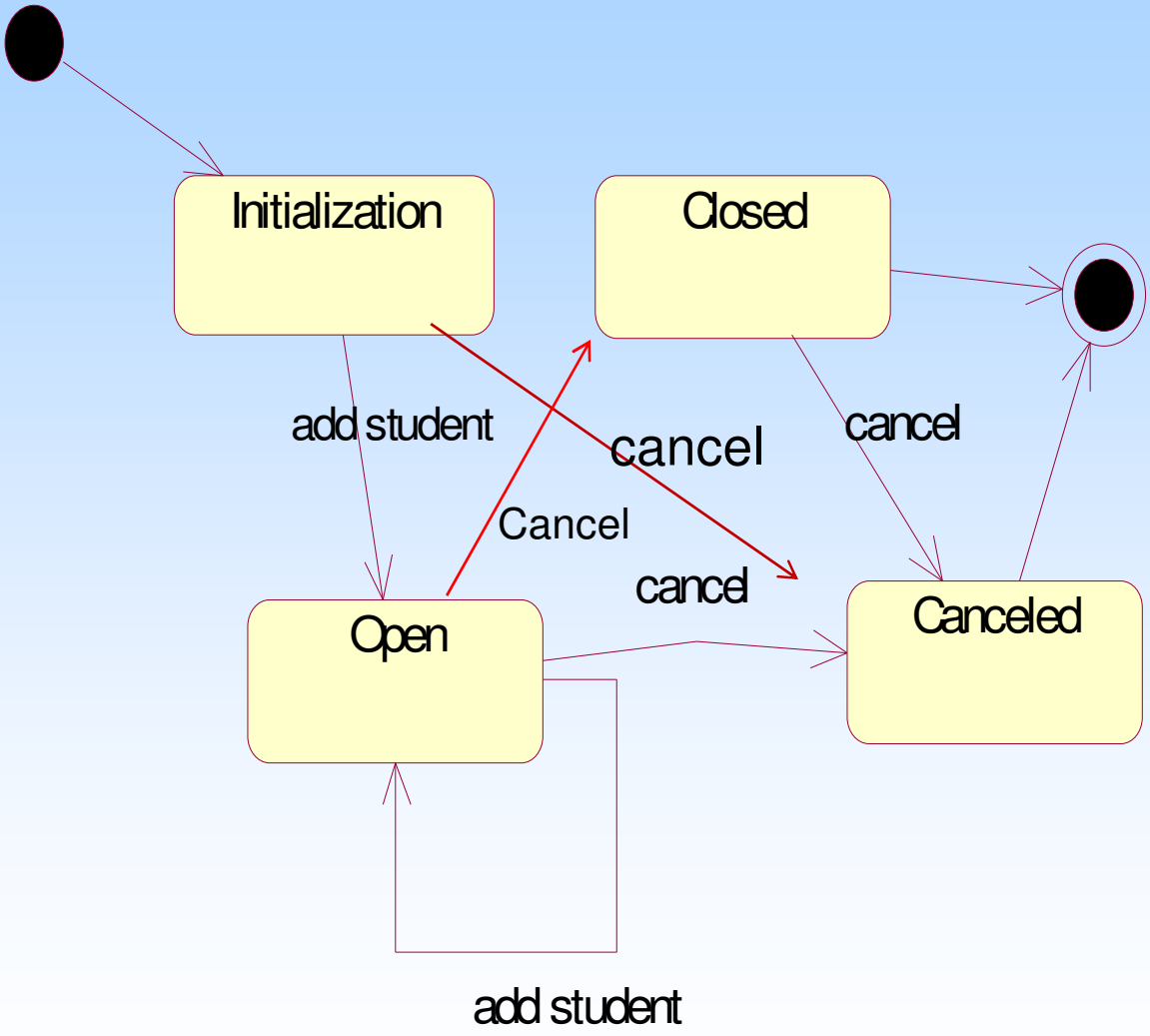
- There are two special states that are added to the state transition diagram.
- **Start** state – Each diagram must have one and only one start state.
- **Stop** state – An object can have multiple stop states.

Start State Stop State

State Transition Diagram –Hotel Room Class



State Transition Diagram– Course Class



State Transition Details

- A State Transition may have the following associated with:
  - an action and/or

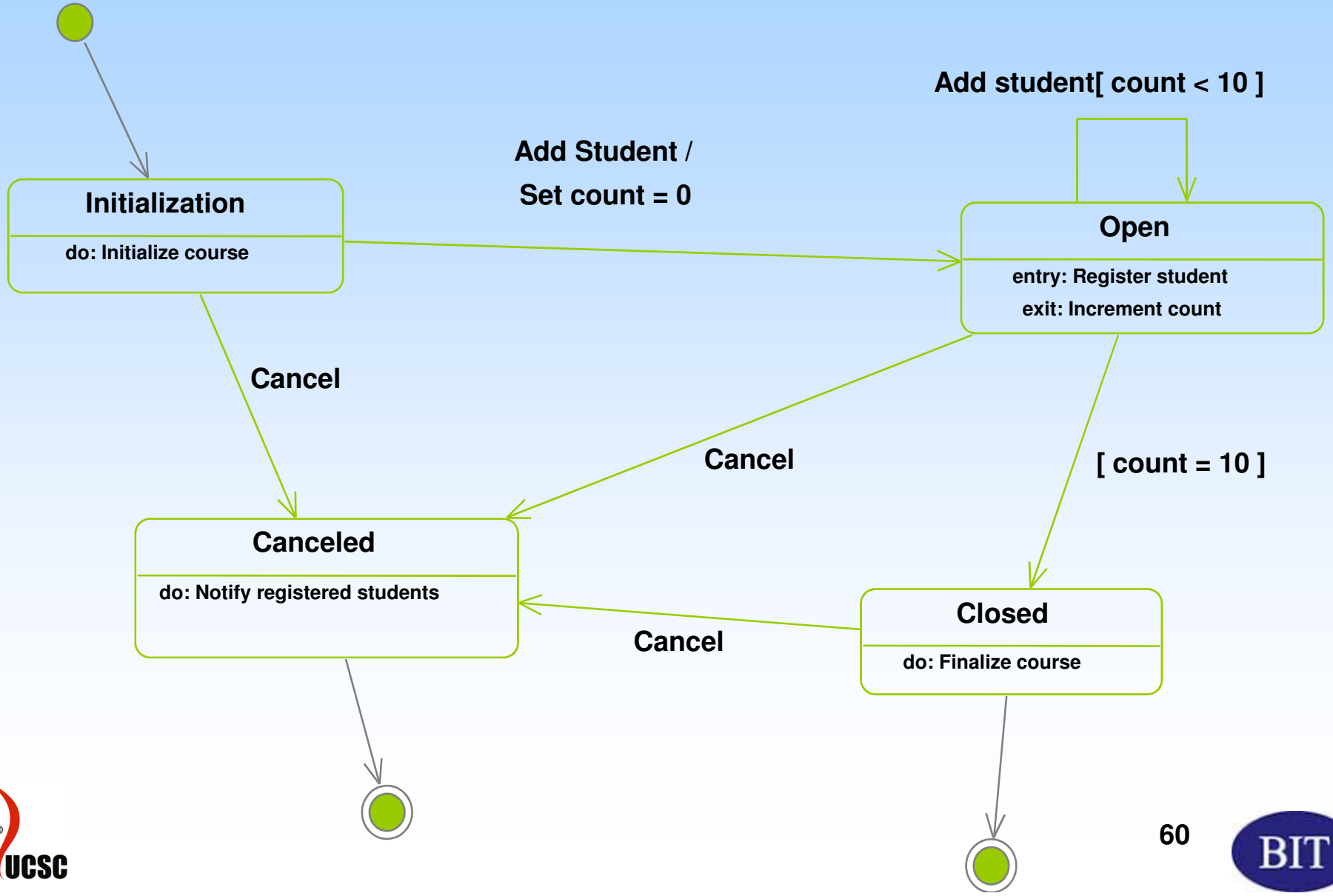
    (behaviour that occurs when the state transition occurs.)
  - a guard condition

    (allows state transition only if it is true.)
- A State Transition may also trigger an event

  A message that is sent to another object in the system.

State Transition Diagram

Course Offering with State Details



State Details

- ***Activity*** : behaviour that an object carries out while it is in a particular state.
  - An activity is shown inside the state itself, preceded by the word *do* and a colon.
- **Entry Action** :
  - Behaviour that occurs while the object is transitioning into the state.
  - Shown inside the state, preceded by the word *entry* and colon.

State Details cont…

- **Exit Action :** occurs as part of the transition out of a state.
  - Shown inside the state, preceded by the word *exit* and colon.
- The behaviour in an activity, entry action, or exit action can include sending an event to some other object.

State Details con…

- In this case, the activity, entry action, or exit action is preceded by a **^**

  Do:^Target.Event(Arguments)

  Target - object receiving the event

  Event - message being sent

  Arguments – parameters of the message being sent

  Eg.

  Do:^CourseRoster.Create

I wish you all the success for your exam