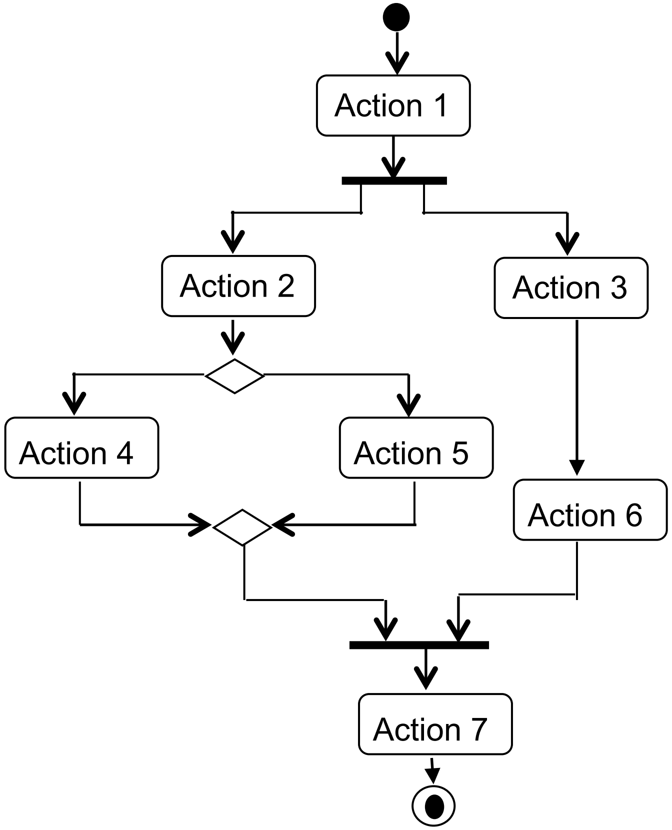
LECTURE 8

Activity Diagram

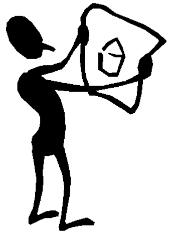
Workflow and Use Case Modeling
with
Activity Diagrams

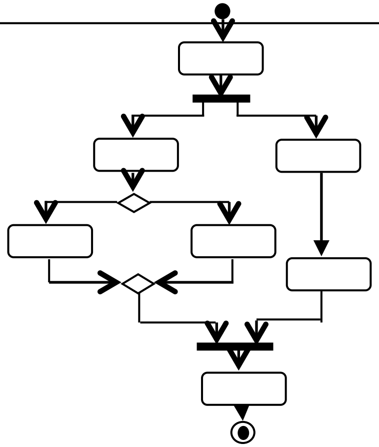
Aims and Learning Outcomes

- **To be familiar with Activity Diagrams**



What is an Activity Diagram?





Activity diagrams are a technique to describe

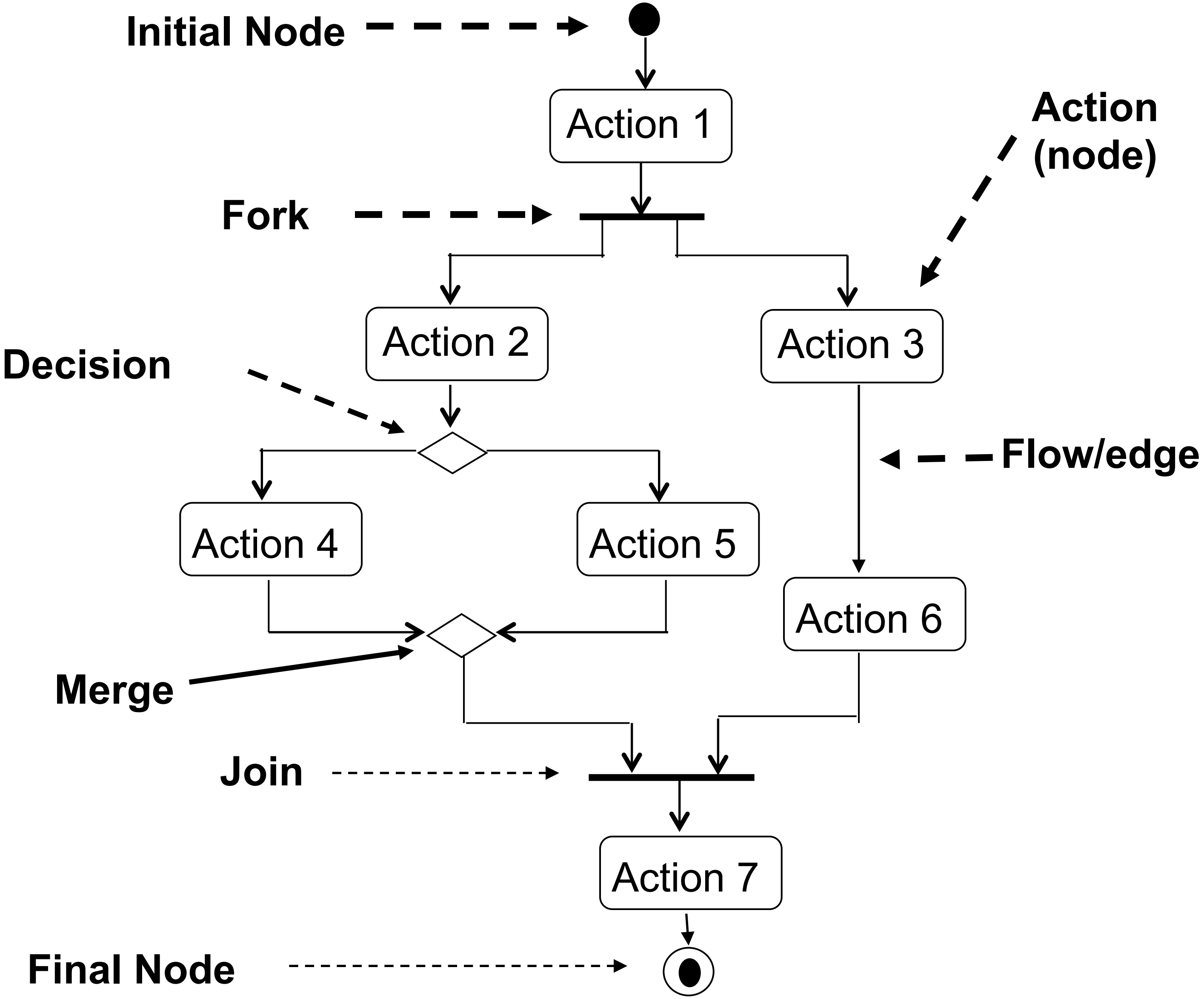
--- procedural logic

--- business processes

--- work flow

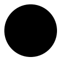
Activity diagrams support parallel behaviour.

Activity Diagrams - Basic

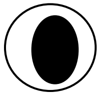


Activity Diagrams - Basic

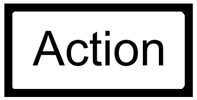
Activity Diagram Elements:



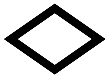
- **Initial node** indicates the start of a flow of activities.



- **Activity final node** indicates the end of an activity



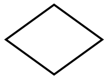
- **Action** describes a basic process or transformation that occurs within a system.



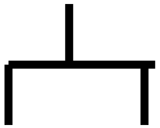
- **Decision** indicates a point of conditional progression:
  - if a condition is **true**, then processing continues one way
  - if a condition is **false**, then processing continues another way

Activity Diagrams - Basic

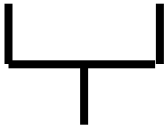
Activity Diagram Elements (contd):



- **Merge** indicates a point that multiple alternative flows are merged to form one flow.

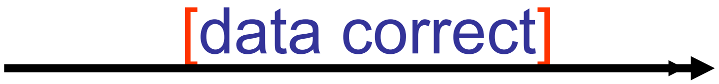


- **Fork** splits one incoming flow into several outgoing concurrent flows

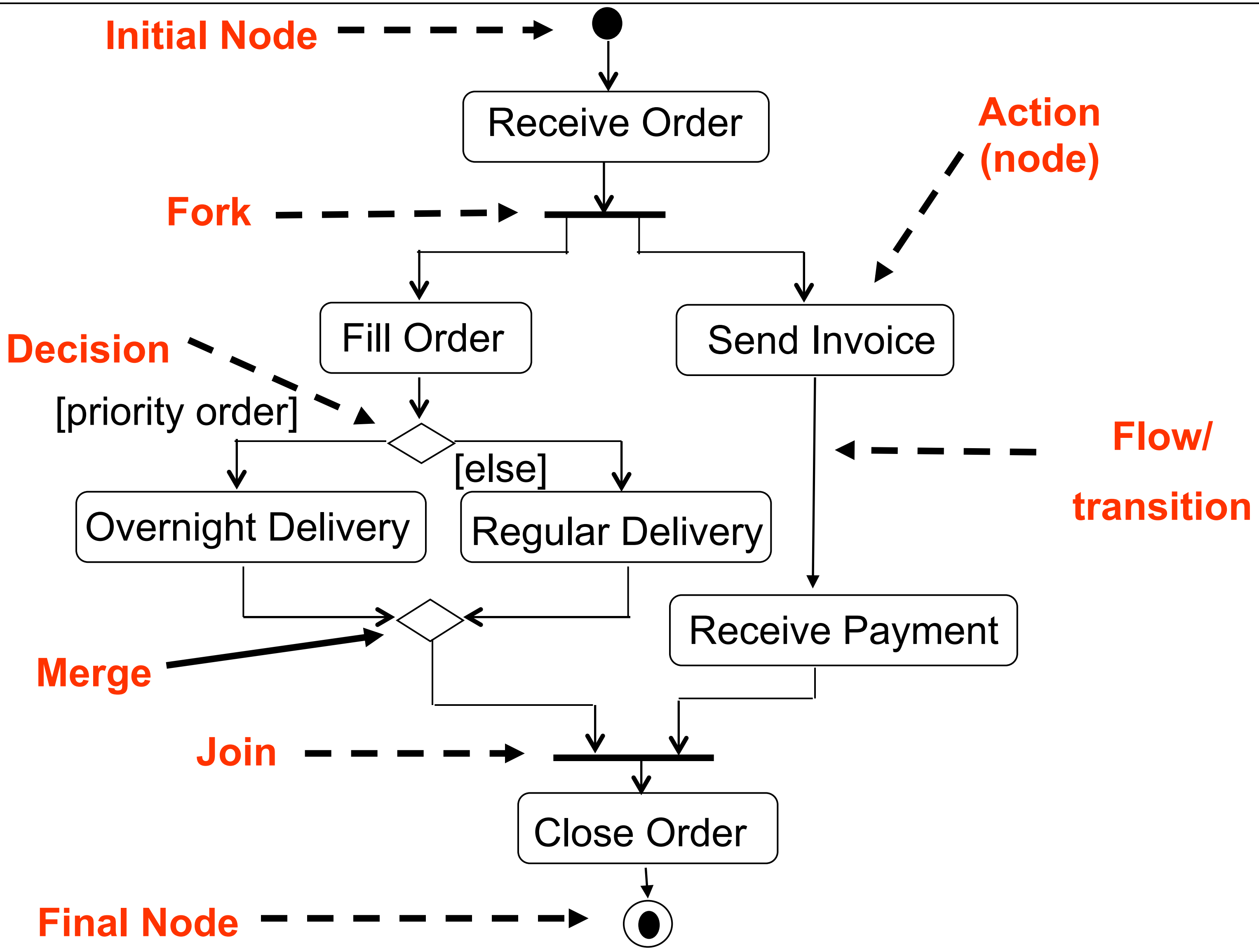


- **Join** ends several incoming concurrent flows into one outgoing flow

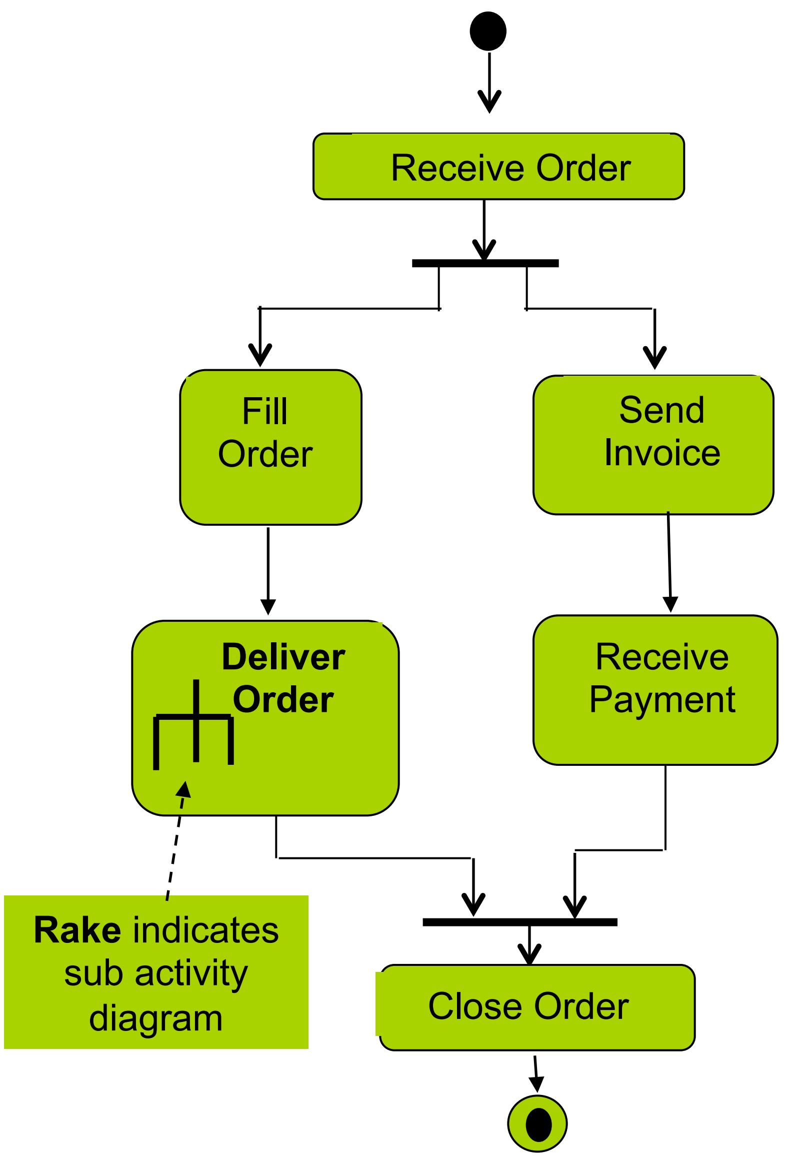
- **Flow/transition** directs the flow of activity from a source node to a target node
- **Guard [ ]**shows a condition that must be true, for a transition to occur.



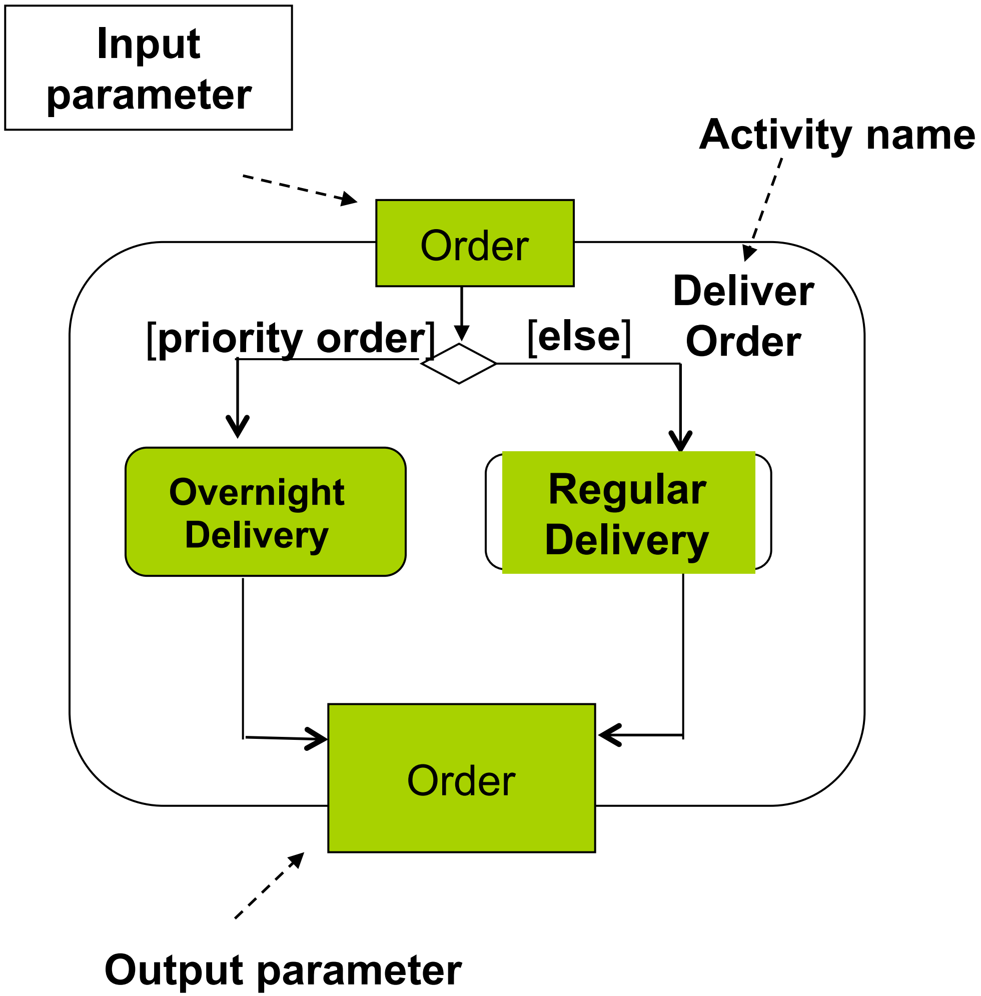
Activity Diagrams-Examples
Example 1: Business Process Modeling



Activity Diagrams-Examples
Example 1 (contd.): Decomposing an action



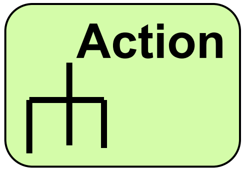
Rake indicates sub activity diagram



Input parameter

Output parameter

Activity Diagrams- Action Decomposition



Rake symbol in an action implies presence of sub-activity diagram for the action.

Action Decomposition is good for

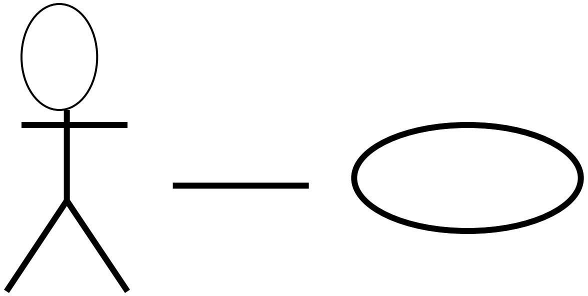
- **D**ecomposing activities from higher, general level business processes to lower detailed processes
- Making a model easier to understand
  - higher level (more abstract) view
  - lower level (more detailed) view

Purpose of Activity Diagrams

1. To allow the reader to see how a system executes
2. To show how the system changes direction based upon different conditions and stimuli

When do you use Activity Diagrams?

- In __________ Phase



- Used to check Use Cases
- In doing so, discover more use cases.
- Remember -- UML is an ___________ process

Why Model a Use Case as an Activity Diagrams?

- Model the workflow of a use case
  - Show paths within the use case
  - Show paths between use cases

- Identify the pre-conditions and the post-conditions that must be met by use cases

How to Model Activity Diagrams

- Five tasks to do:

1. Identify the ______case to model
2. Model ________ path for each use case
3. Model _______ paths for each use case
4. Add swimlanes
5. Iterate – refine high-level activities into more ________ diagrams

Activity Diagrams-Examples
Example 2: Use case modeling

Text based use case: **Buy a product**

Main Success Scenarios (MSS)

1. Customer selects items to buy
2. Customer fills in shipping information
3. System presents full price information, including shipping
4. Customer fills in credit card information
5. System authorises purchase
6. System confirms sale

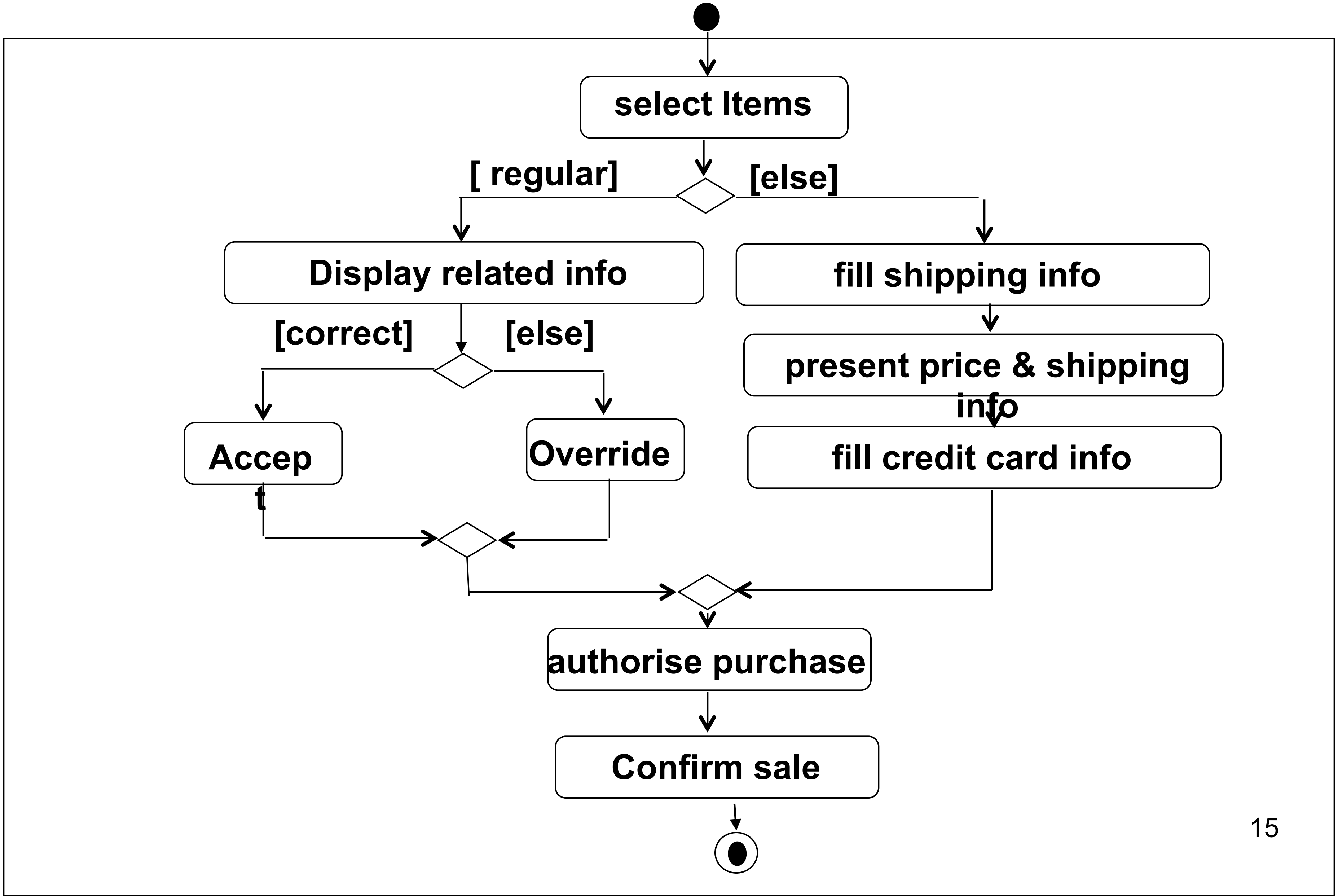
Extensions:

2.a. Customer is regular customer

.1: System displays current shipping and pricing information

.2: Customer accept or override the above, return to MSS at step 5

Example 2 (continue): Without Partition



select Items

[regular]

[else]

Display related info

fill shipping info

[correct]

[else]

present price & shipping info

Accep t

Override

fill credit card info

authorise purchase

Confirm sale

Activity Diagrams- Partition

Why partition:

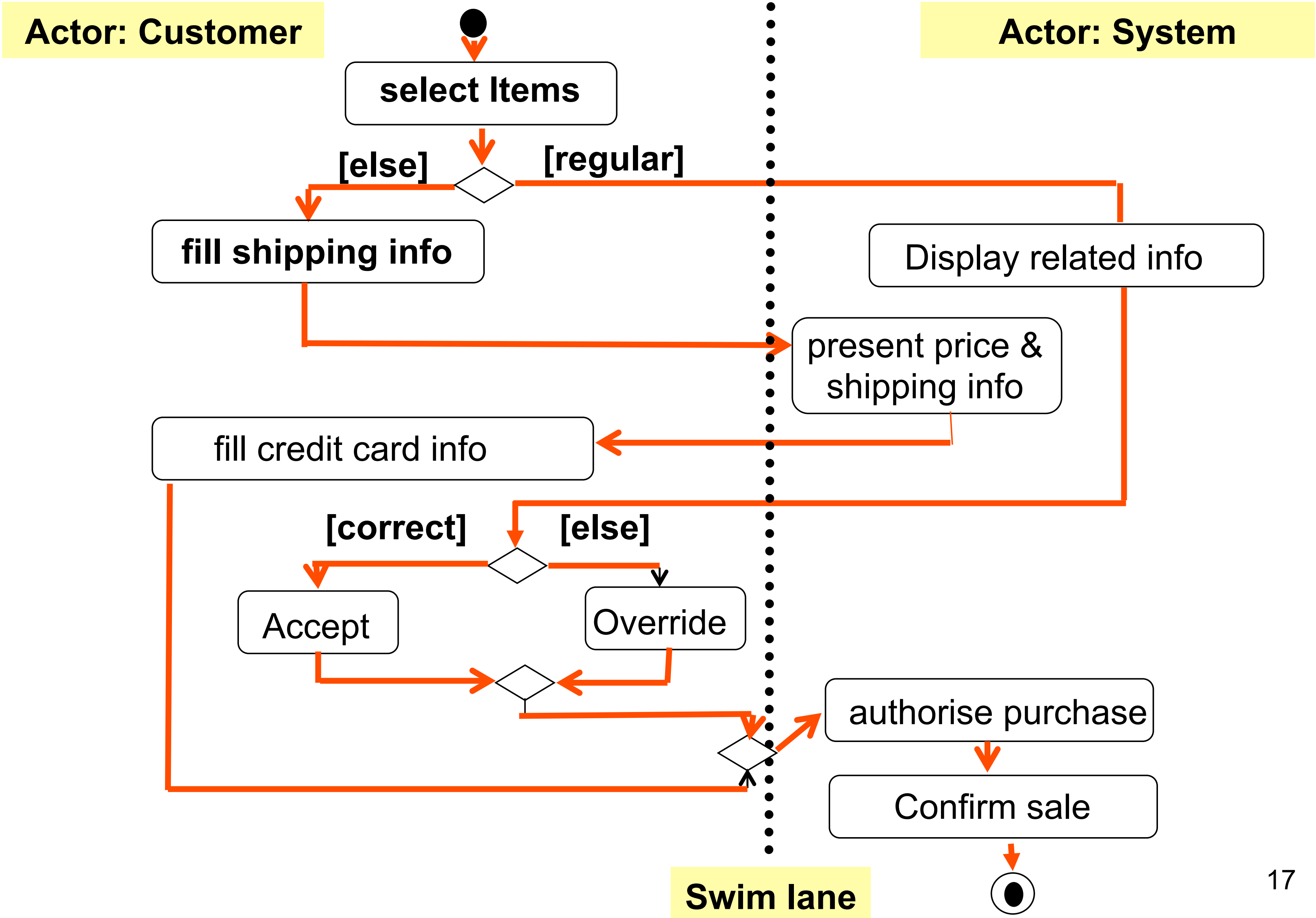
Activity diagrams, without partitions, tells you w_____ happens but do not tell you w____ does what.

With partitions, activity diagrams tells you not only w_____ happens but also tell you w_____ does what.

How to partition:

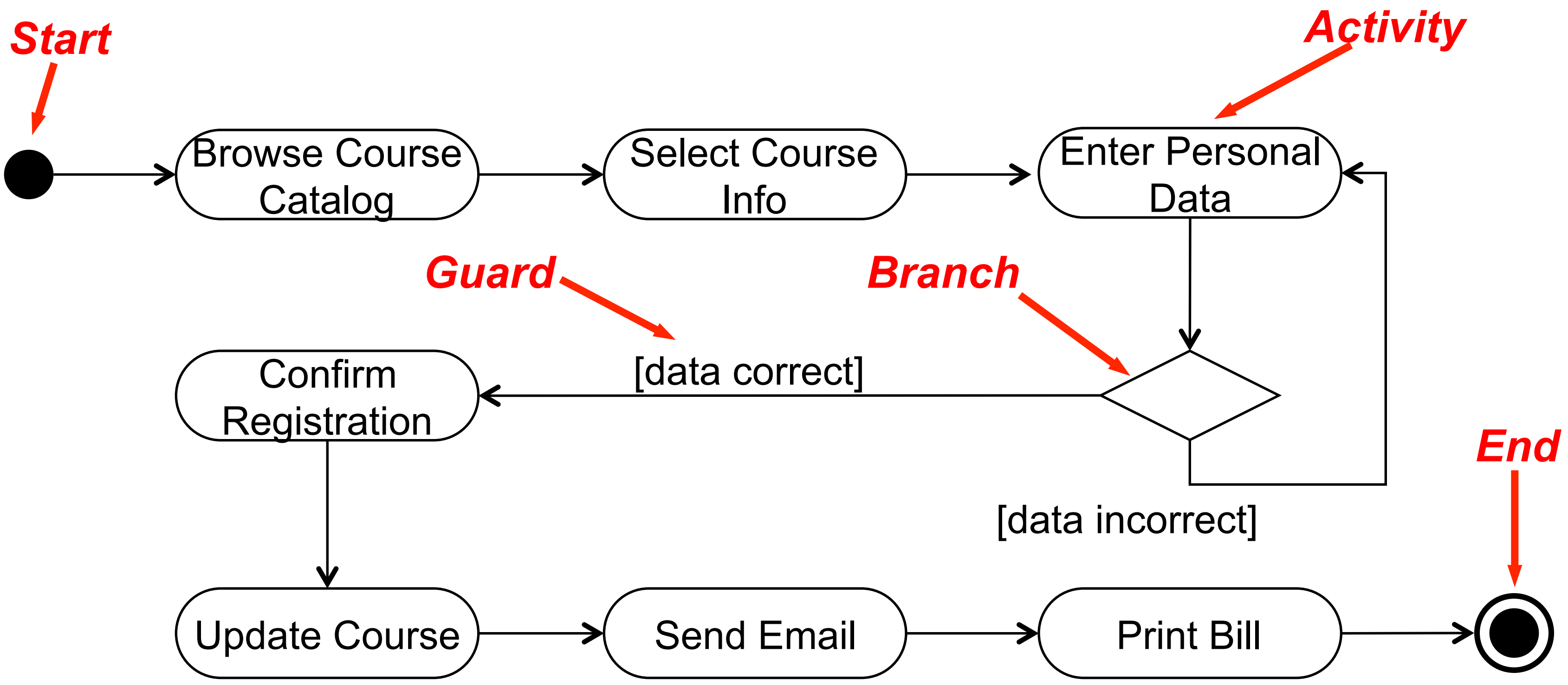
- Separate the diagram into parallel lanes called **swimlanes**
- Each _______ shows the name of the actor at the top, and presents the activities of each ________.

Example 2 (continue): With Partition



Example 3 --- Register for a Course

- Course Registration

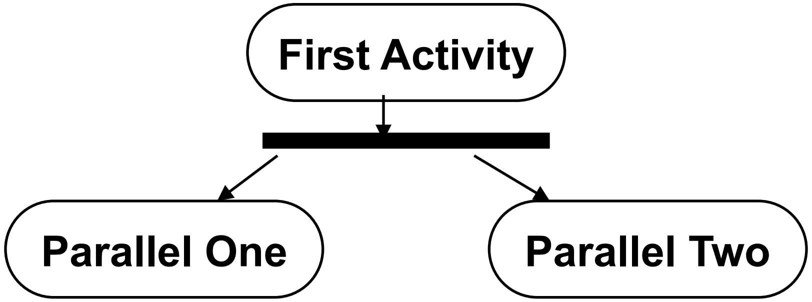


Modeling Parallel Behaviour

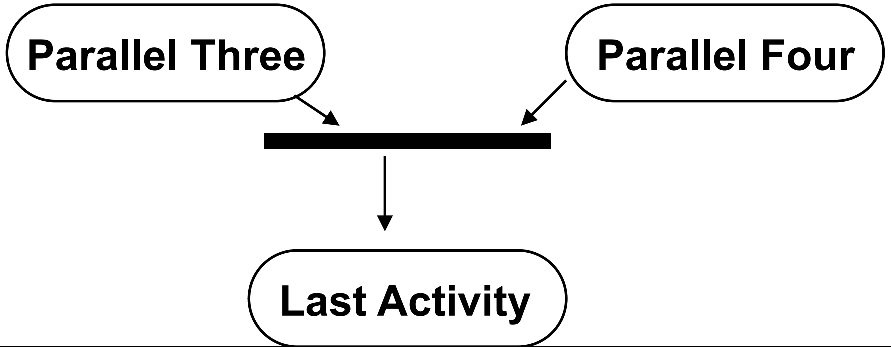
- When we look at the diagram, can see that several actions could be executed in p_____________.
- E.g. Send Email could be executed while the course is updated and while the bill is printed.
- Activity diagrams are good at showing p________ behavior
- Redraw the diagram with explicit modeling of the concurrent behavior.
- Shown in the next diagram.

Forks and Joins

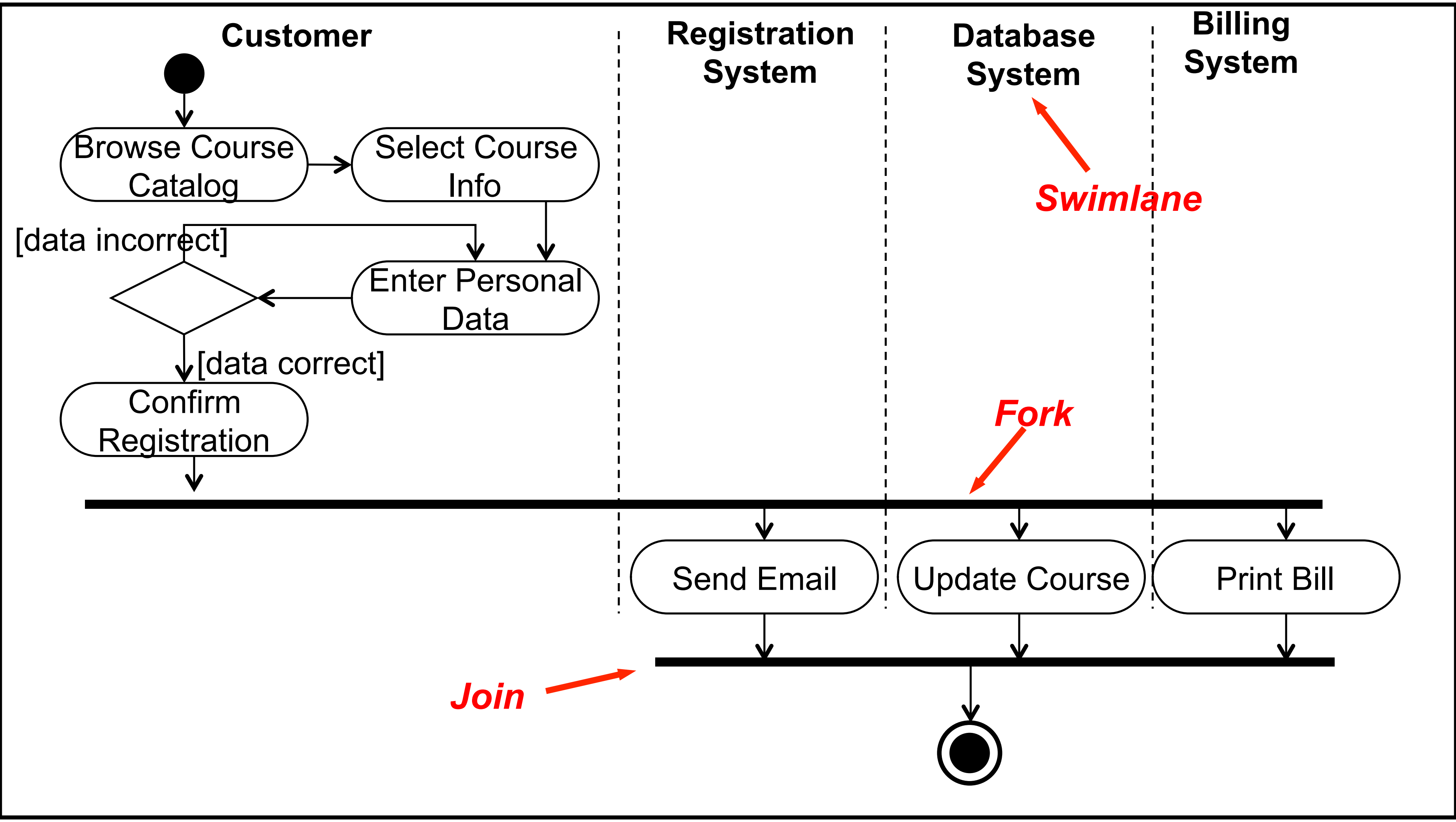
- Used with parallel processes
- Fork has one i_______ transition & several o_______ transitions.
- Output transitions are all executed in p_______.



- Join has several i___ transitions & only one o___ transition.
- Output transition is only executed when all i_____ transitions have completed their activity.

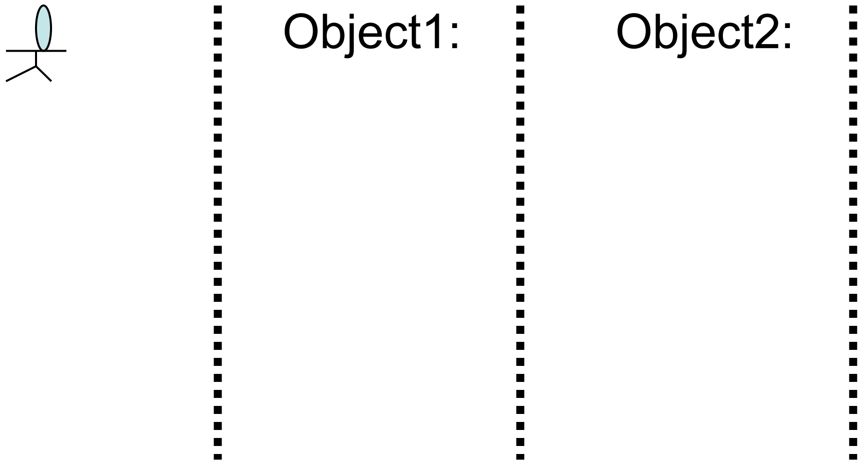


Example 3 – with Parallel Activities



Swimlanes

- Large rectangular boxes with the name of the object or domain at the top
- Allow you to specify who is doing a particular action.
- For this, you arrange your actions in vertical zones, each separated by dashed lines

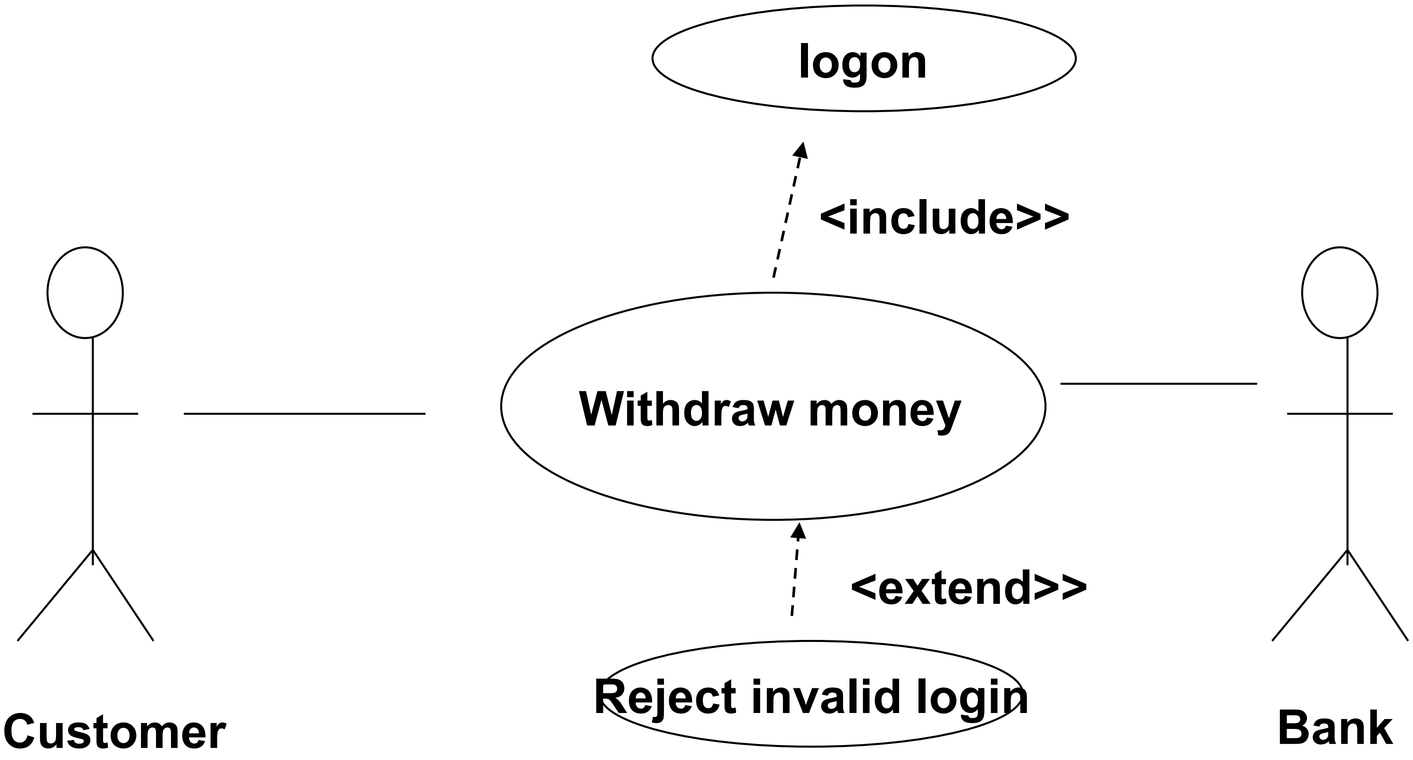


Swimlanes

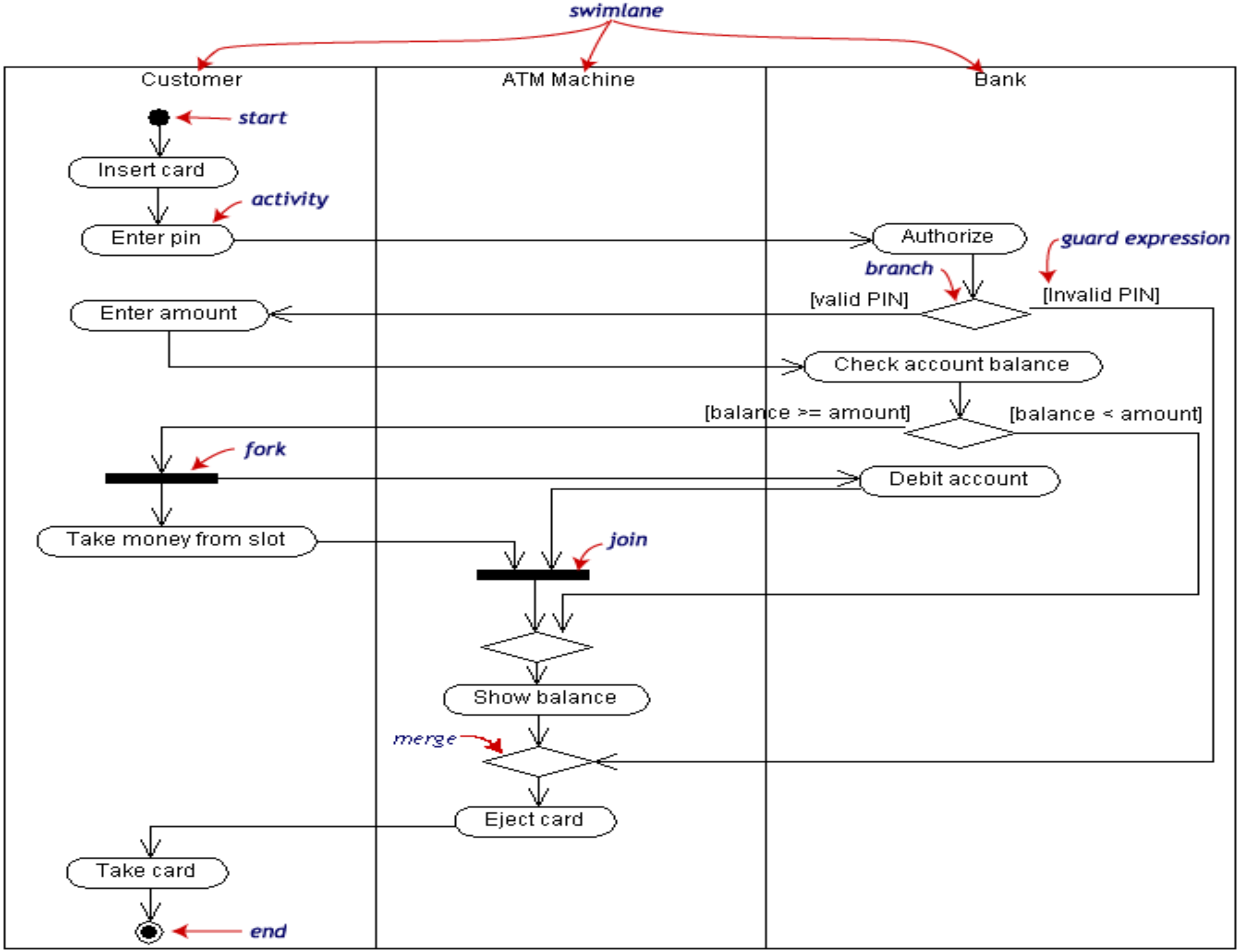
Advantages

- Can show activities done by various actors in a use case diagram
- Can used for domain modeling.
- Each zone can represent a user, a department, an existing system etc. that is known to execute the action, but not yet defined as a class.
- Means, you are still modelling in the user's world of business, not in OO world
- Increase the readability of Activity Diagrams

Use Case – Withdraw Money

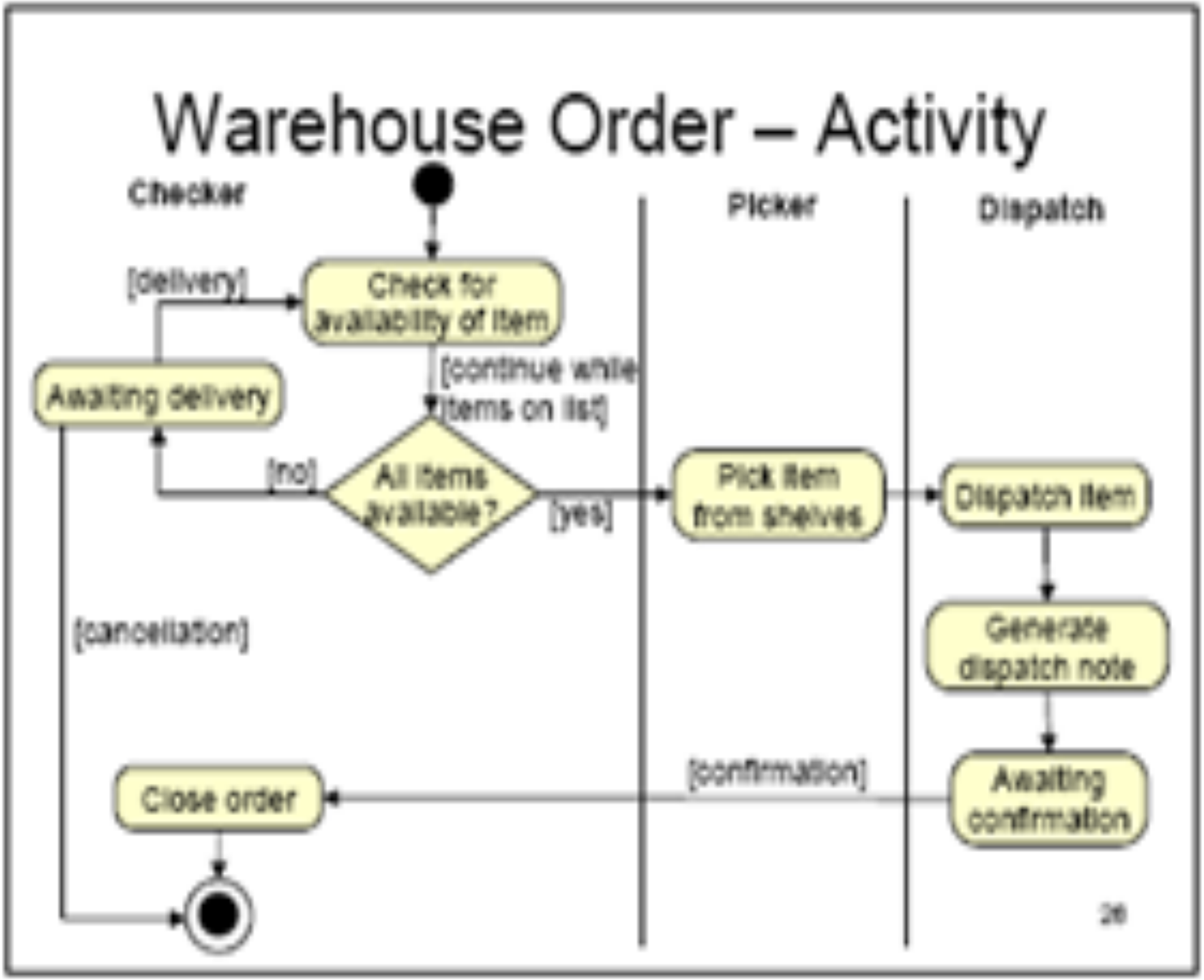


Example 4: Activity Diagram to Withdraw Money from Bank A/c using ATM



Example 5:Warehousing System

- Orders are received in the Ordering Department and items on the orders are checked for availability.
- If all items are available, the order is passed to the Picking Department for picking the physical items from the shelves and then onto the Dispatch Department from which a despatch note will be generated.
- If some items are not available, then the order goes back to the Ordering Department until the missing items are delivered. Then it is passed onwards, as described above.
- If the missing items take too long to come in, the customer may cancel the order.
- When the order is finally delivered, the Ordering Department is informed so that the order can be closed.



Warehouse Order – Activity

Checker

Picker

Dispatch

[delivery]

Check for availability of item

[continue while items on list]

Awaiting delivery

[no]

All items available?

[yes]

Pick item from shelves

Dispatch item

Generate dispatch note

[cancellation]

[confirmation]

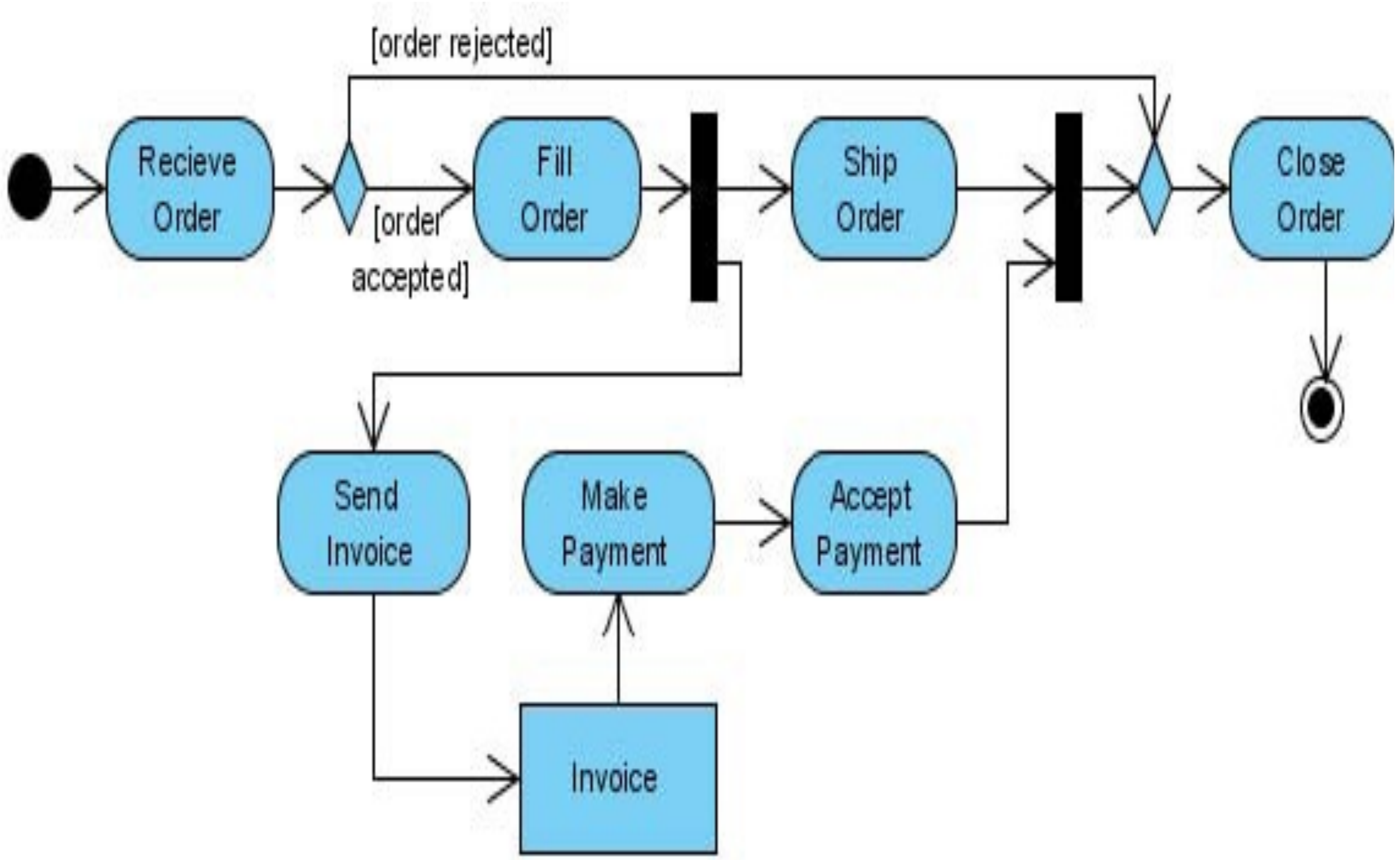
Awaiting confirmation

Close order

Example 6: Placing an Order

- A customer can buy books by calling the Sales Department of XYZ Book Shop by phone and providing Sales with order details.
- Once Sales receive the order, a clerk will check for the membership status of the customer.
- If a customer's membership is still valid, Sales will enter the order details, as given by the customer.
- Sales will then pack the items, await for shipment and, at the same time, create an invoice which they will send out.
- If Sales has received a cheque from the customer, the cheque will be lodged in the bank account.
- If the cheque is settled with the bank, then Sales will arrange the shipment for the customer.
- The order will then be completed.

Example 6: Placing an Order



Activity Diagrams- Summary

Activity Diagrams can be used to model and design

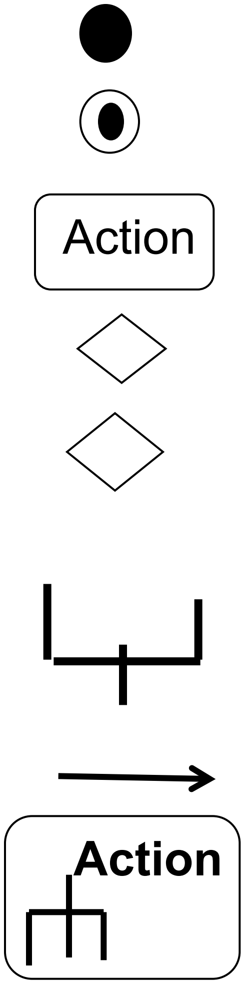
- Business process
- Workflow
- Complex use case flows
- Procedural logic
- Algorithms

Activity Diagrams- When to use

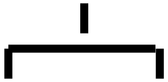
Activity diagrams are most versatile UML diagram and can be used at the different levels of system design including

- Business process
- Workflow
- Use case flows
- Procedural logic
- Algorithms
- Show parallel business flows

Activity Diagrams –Summary



Activity Diagram Elements:

- Initial node
- Activity final node
- Action
- Decision
- Merge
- Fork 
- Join
- Flow/Transition
- Action decomposition

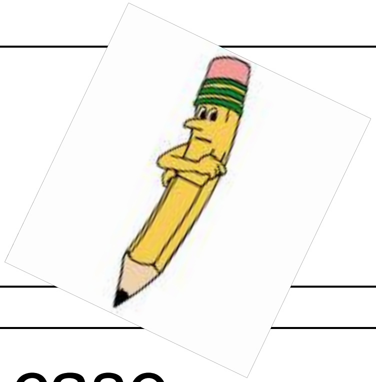
When to Use

- Analyzing Use Cases
- Understanding workflows
  - Good visualising diagrams
- Describing complicated sequential algorithms
  - Not yet software related
  - Good for showing to users and customers
- Modeling parallel behaviour

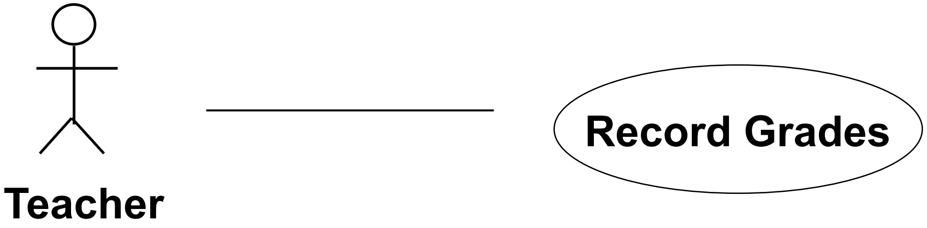
Advantages of Activity Diagrams

- Don’t require technical expertise
- Don’t need to be an expert on OOAD
- All types of users can understand them
- Can show varying level of complexity

Exercise 1



- Draw an Activity Diagram for this use case



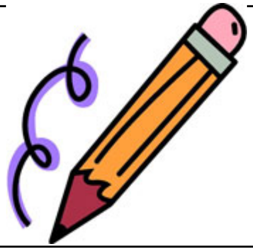
- On further analysis, there are 3 steps necessary for the teacher to record grades (select a student, enter grades, and save grades).

Exercise 2

Alter your Activity Diagram to consider the following alternate paths -

- What if no grades exist for a student?
- What if grades already exist for a student?

Exercise 3



Draw an Activity Diagram showing swimlanes to show the following:

A teacher logs onto a web site, which validates the user.

- An error message is displayed, if the user is invalid.
- The teacher enters a student’ s name, information on that student is retrieved from a database and displayed.
- The teacher update the student’ s grades, and this information is stored permanently.
- If there is a problem saving the student information, an error message appears on screen.

Exercise 4

Draw an activity diagram to represent the following scenario:

Three days before the flight, my travel agent emails me with a list of required travel documents. If the list is not received by the three day deadline, I cancel the flight. Otherwise: Three hours before the flight, I order a taxi. When the taxi arrives, I leave for the airport.

Exercise 5

Question 5

Explain how the following activity diagram is implemented.