Object-Oriented Design

Lecturer: Raman Ramsin

Lecture 11:

Activity Diagrams – Part 1

Analysis Workflow: *Analyze a Use Case*

- The *analysis workflow* consists of the following activities:
  - Architectural analysis
  - **Analyze a use case**
    - **Outputs:**
      - **analysis classes**
      - **use case realizations**
  - Analyze a class
  - Analyze a package

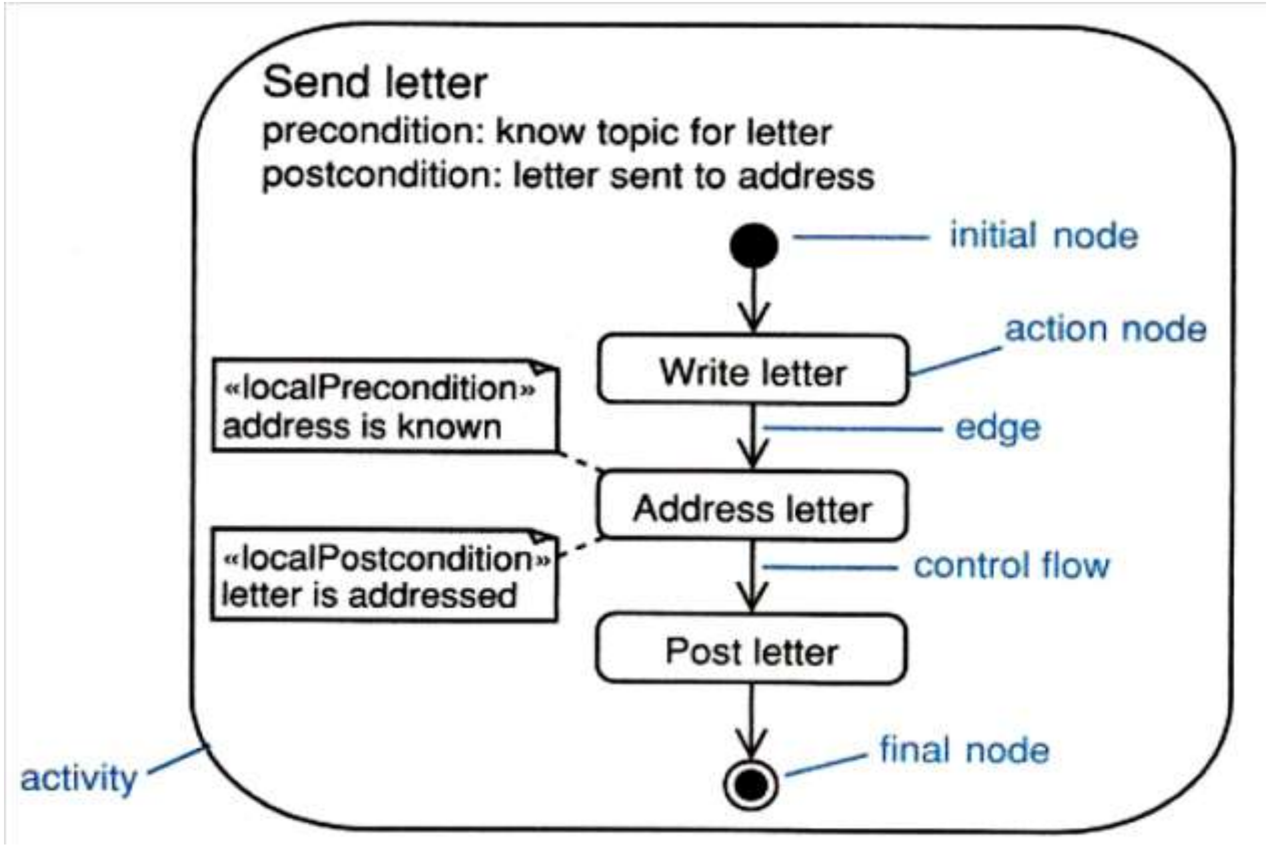
Activity Diagrams

- Activity diagrams are OO flowcharts:
  - used for modeling all types of processes;
  - can be attached to *any* modeling element to capture its behavior;
  - a good activity diagram communicates one specific aspect of a system's behavior;
  - in UML 2, activity diagrams have Petri Net semantics.

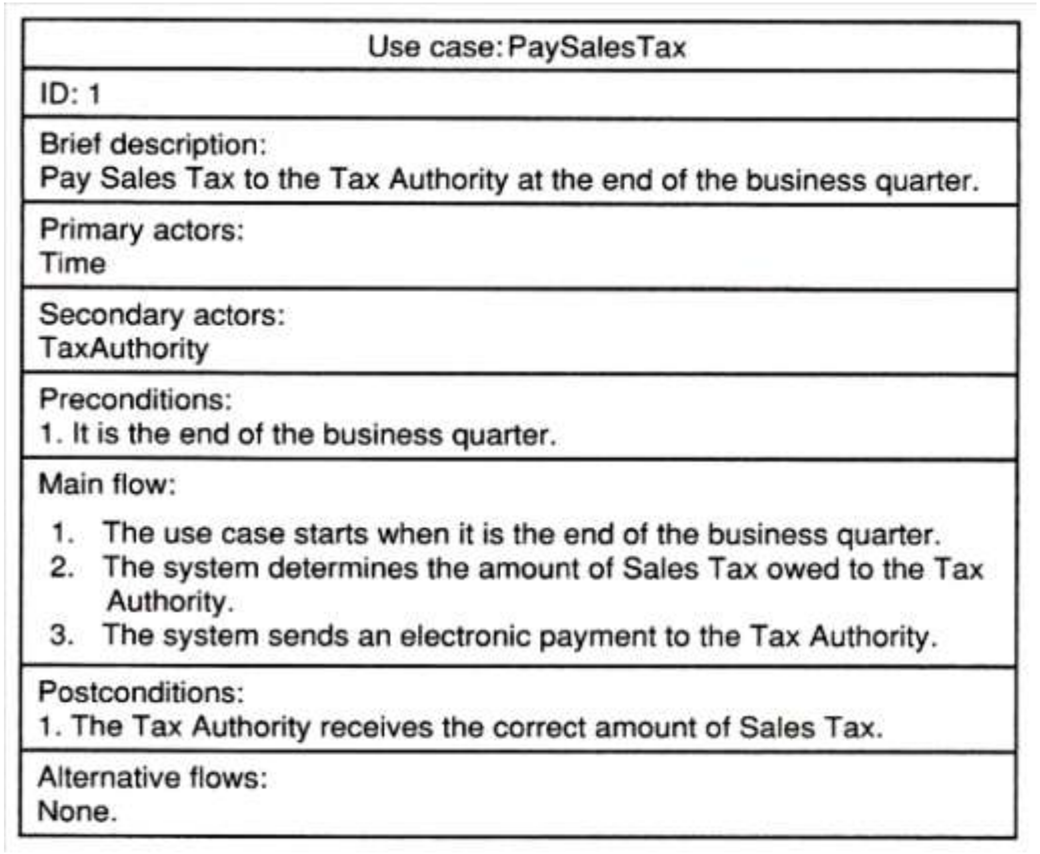
Activities

- Activities are networks of nodes connected by edges.
- Categories of nodes:
  - **action nodes** - atomic units of work within the activity;
  - **control nodes** - control the flow through the activity;
  - **object nodes** - represent objects used in the activity.
- Categories of edges:
  - **control flows** - represent the flow of control though the activity;
  - **object flows** - represent the flow of objects through the activity.
- Activities can have preconditions and postconditions.

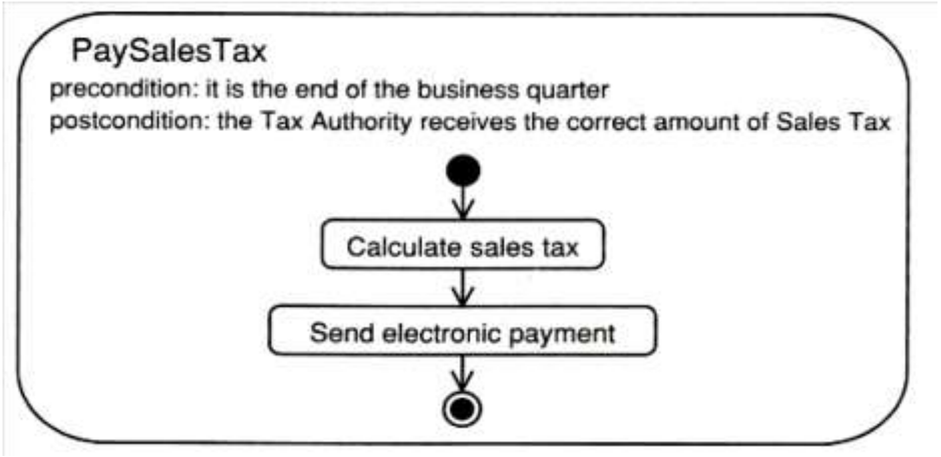
Activities: Example


Send letter
precondition: know topic for letter
postcondition: letter sent to address
initial node
action node
Write letter
edge
«localPrecondition» address is known
Address letter
«localPostcondition» letter is addressed
control flow
Post letter
final node
activity

Activity Diagrams: Use Case Modeling

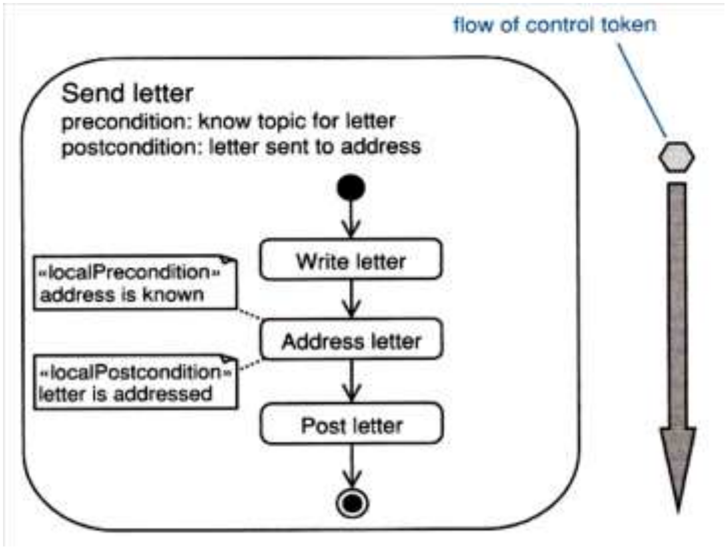


Use case: PaySalesTax
ID: 1
Brief description: Pay Sales Tax to the Tax Authority at the end of the business quarter.
Primary actors: Time
Secondary actors: TaxAuthority
Preconditions: 1. It is the end of the business quarter.
Main flow: 1. The use case starts when it is the end of the business quarter. 2. The system determines the amount of Sales Tax owed to the Tax Authority. 3. The system sends an electronic payment to the Tax Authority.
Postconditions: 1. The Tax Authority receives the correct amount of Sales Tax.
Alternative flows: None.


PaySalesTax
precondition: it is the end of the business quarter
postcondition: the Tax Authority receives the correct amount of Sales Tax
Calculate sales tax
Send electronic payment

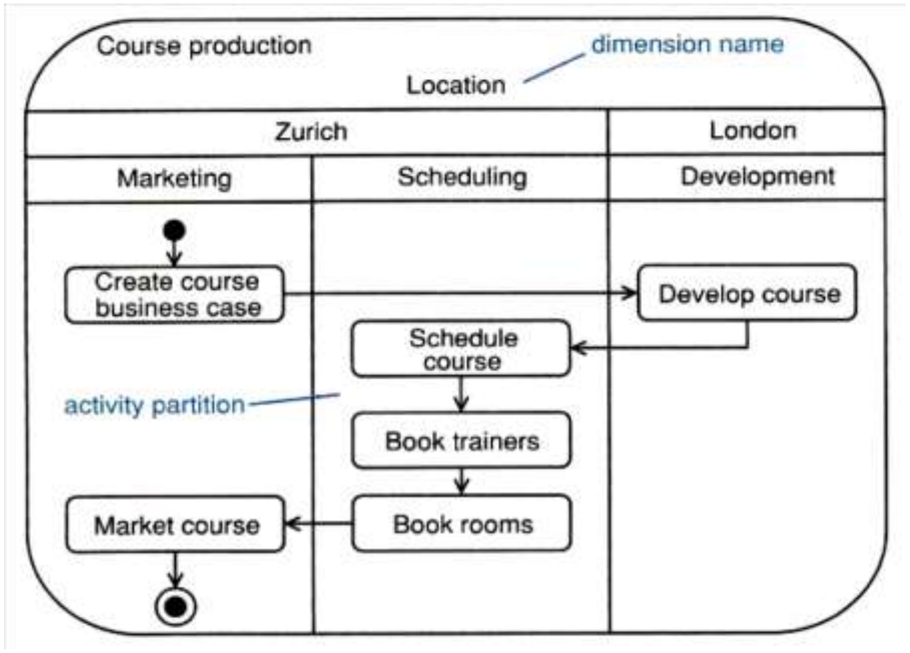
Activities: Tokens

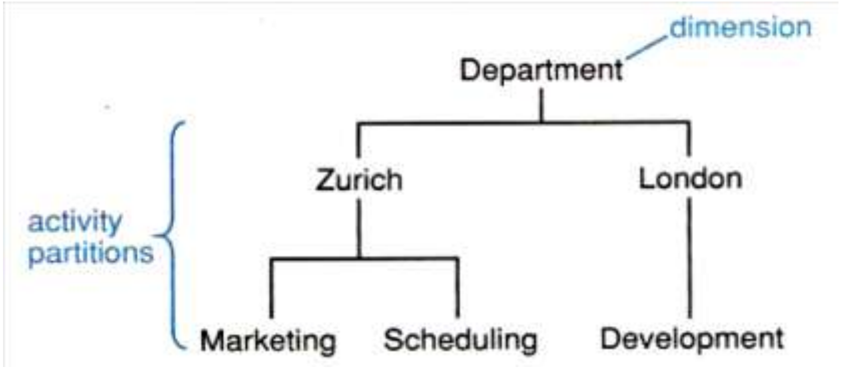
- Tokens flow around the network and can represent:
  - the flow of control;
  - an object;
  - some data.
- Tokens move from a source node to a target node across an edge depending on:
  - source node postconditions;
  - edge guard conditions;
  - target preconditions.

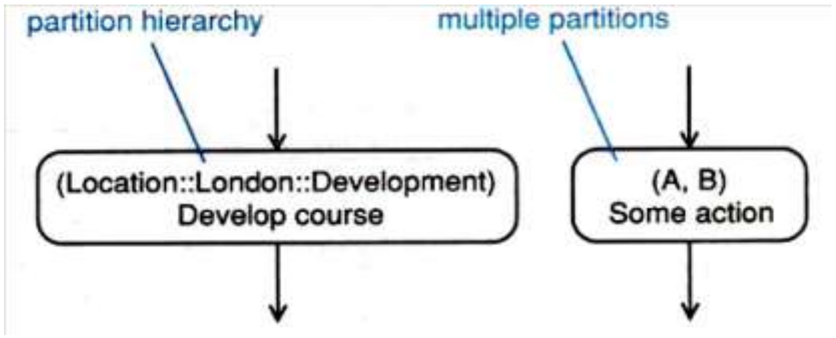

flow of control token
Send letter
precondition: know topic for letter
postcondition: letter sent to address
«localPrecondition» address is known
«localPostcondition» letter is addressed
Write letter
Address letter
Post letter

Activity Partitions

- Activity partitions - a high-level grouping of related actions.
  - Partitions form a hierarchy rooted in a dimension.

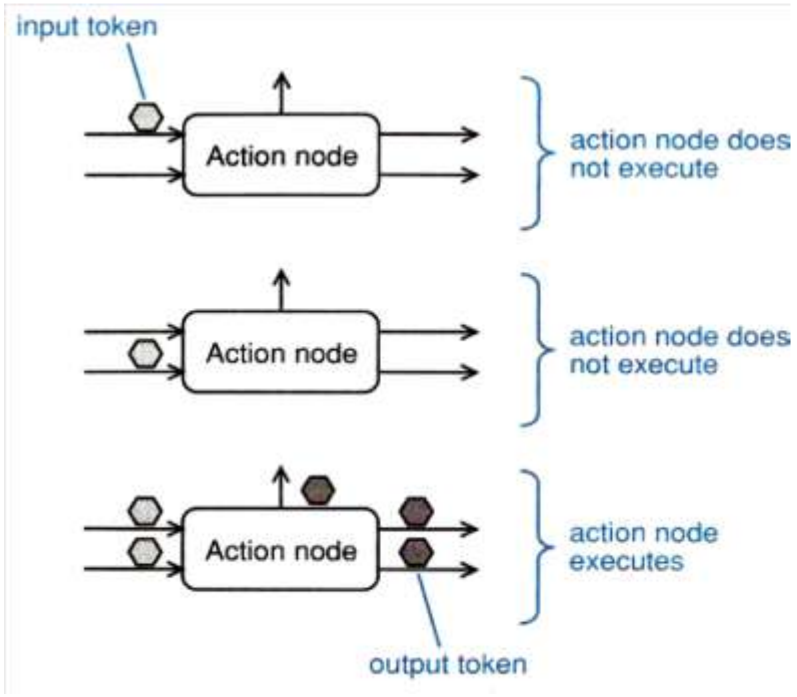


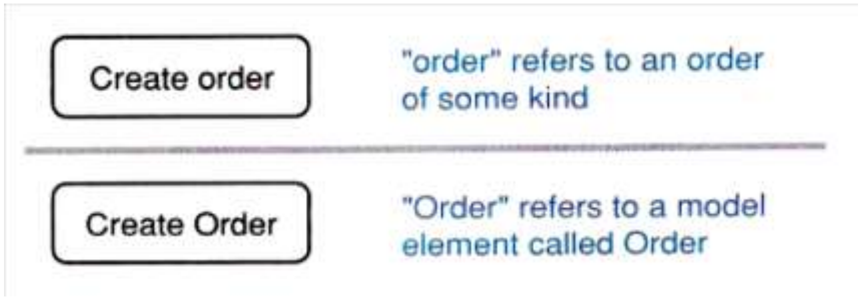




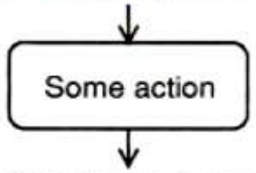
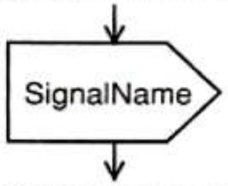
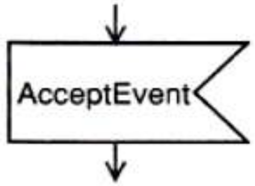
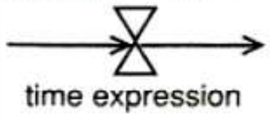
Action Nodes

- Execute when there is a token simultaneously on each of their input edges AND their preconditions are satisfied.
- After execution, action nodes offer tokens *simultaneously* on all output edges whose postconditions are satisfied:



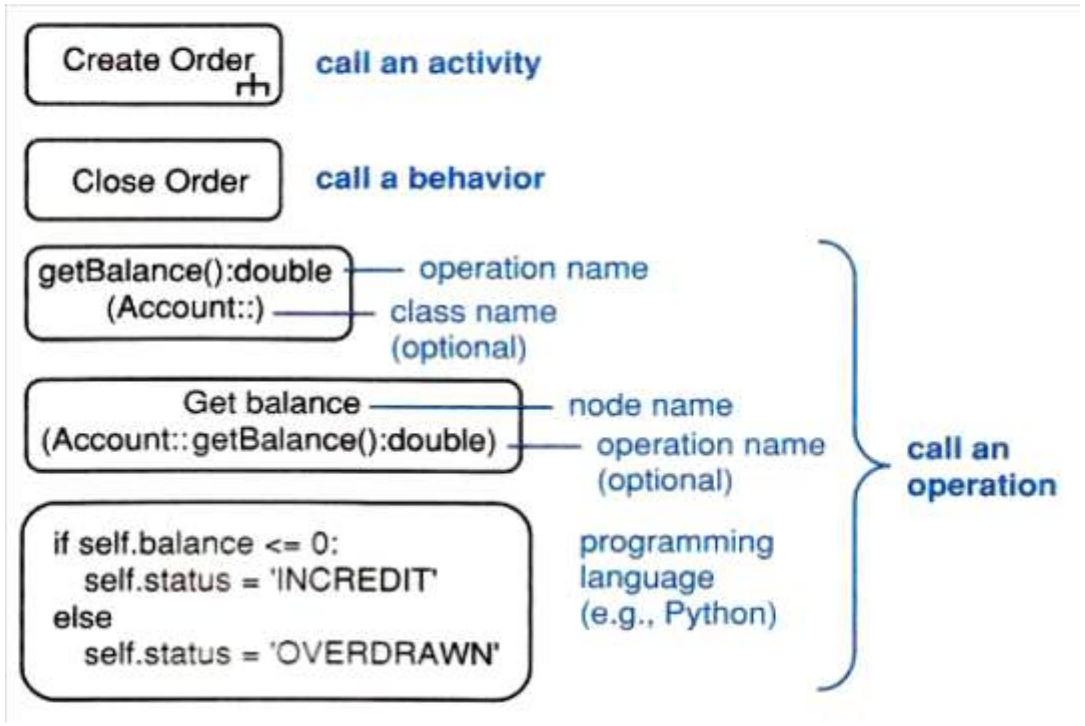

Create order
"order" refers to an order of some kind
Create Order
"Order" refers to a model element called Order

Action Nodes: Types

Syntax	Name	Semantics
	Call action node	Invokes an activity, behavior, or operation
	Send signal	Send signal action – sends a signal asynchronously (the sender *does not* wait for confirmation of signal receipt) It may accept input parameters to create the signal
	Accept event action node	Accepts an event – waits for events detected by its owning object and offers the event on its output edge Is enabled when it gets a token on its input edge If there is *no* input edge, it starts when its containing activity starts and is always enabled
	Accept time event action node	Accepts a time event – responds to time Generates time events according to its time expression

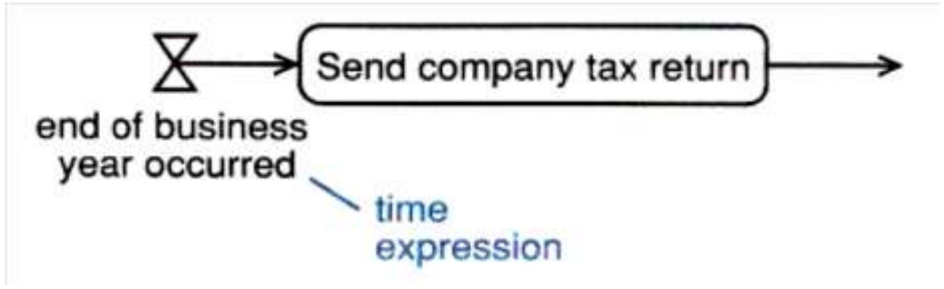
Action Nodes: Call

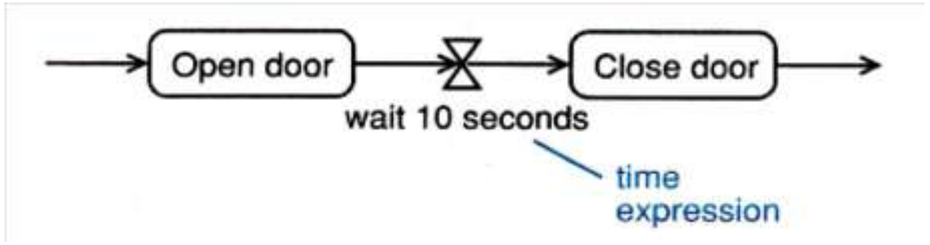
- **Call** action node:
  - call an activity - use the rake symbol;
  - call a behavior;
  - call an operation.



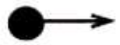
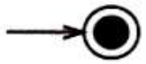
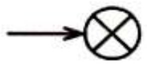
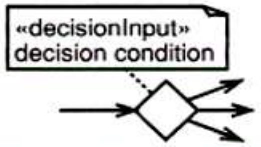
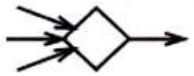
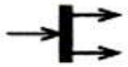
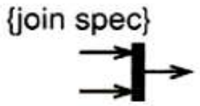
Action Nodes: Accept Time Event

- **Accept time event** action node - executes when its time expression is true:
  - an event in time (e.g., end of business year);
  - a point in time (e.g., on 11/03/1960);
  - a duration (e.g., wait 10 seconds).

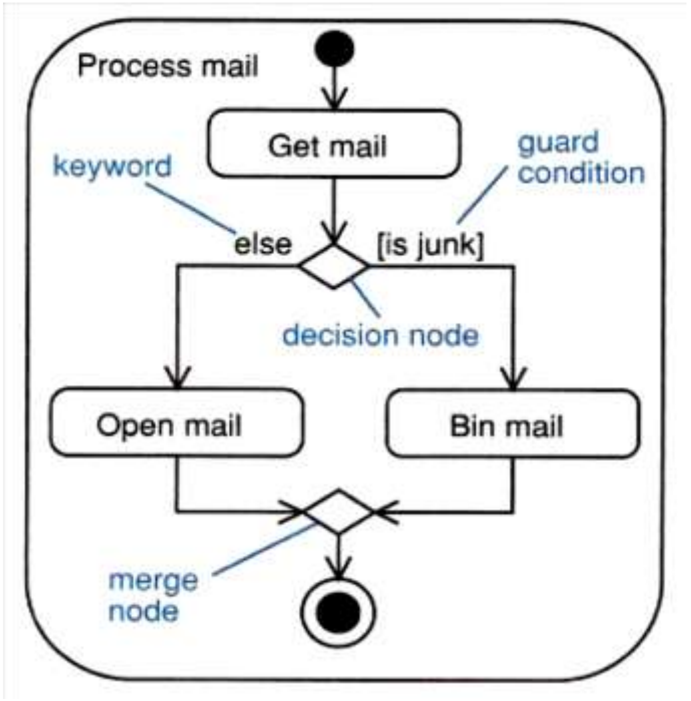


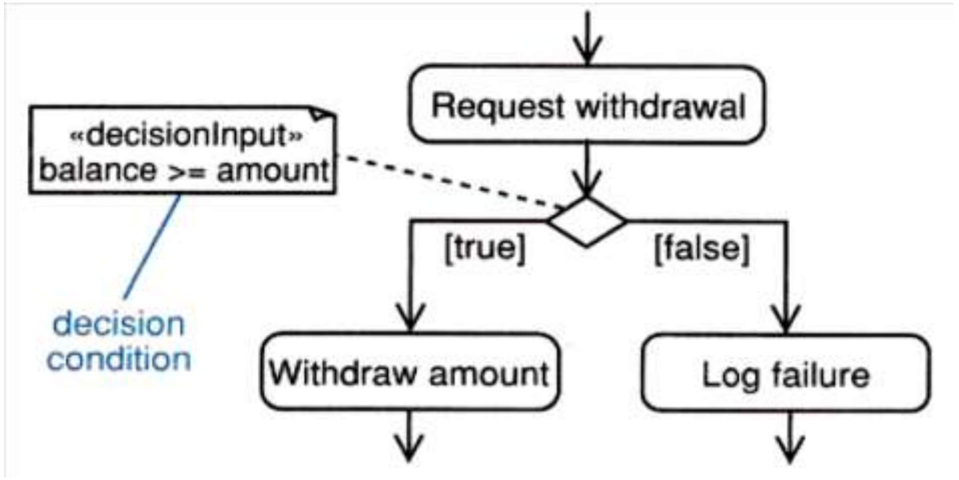


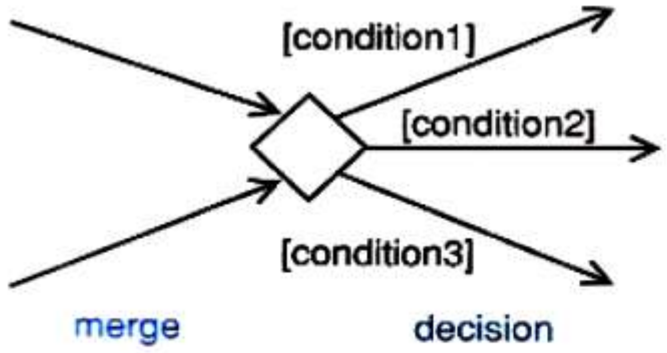
Control Nodes

Syntax	Name	Semantics	
	Initial node	Indicates where the flow starts when an activity is invoked	
	Activity final node	Terminates an activity	Final nodes
	Flow final node	Terminates a specific flow within an activity – the other flows are unaffected	Final nodes
	Decision node	The output edge whose guard condition is true is traversed May optionally have a «decisionInput»	
	Merge node	Copies input tokens to its single output edge	
	Fork node	Splits the flow into multiple concurrent flows	
	Join node	Synchronizes multiple concurrent flows May optionally have a join specification to modify its semantics	

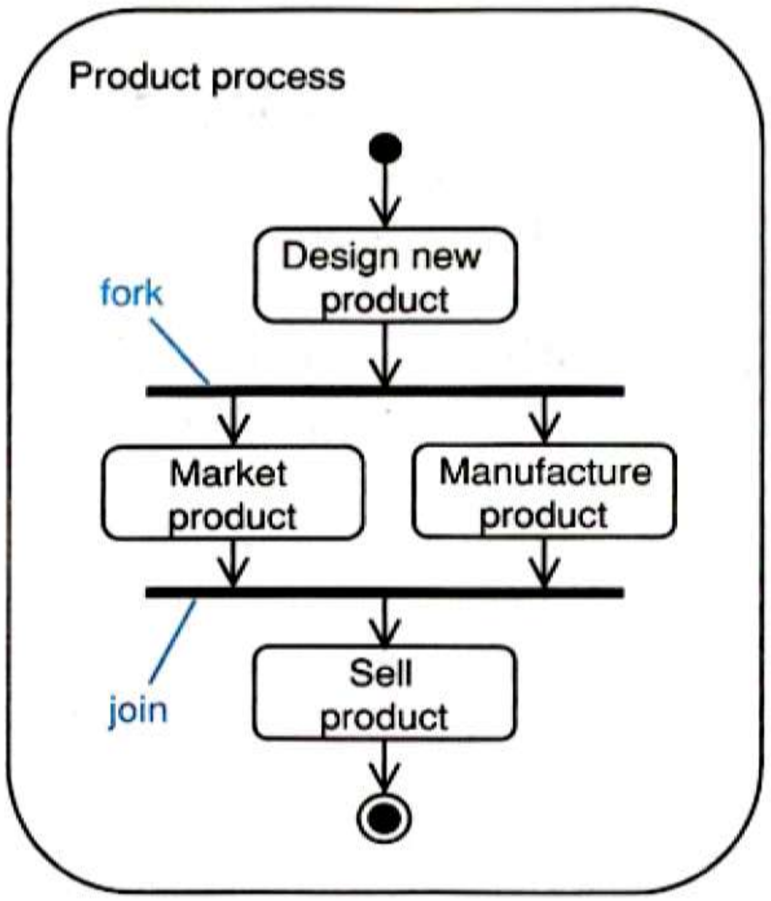
Control Nodes: Decision and Merge





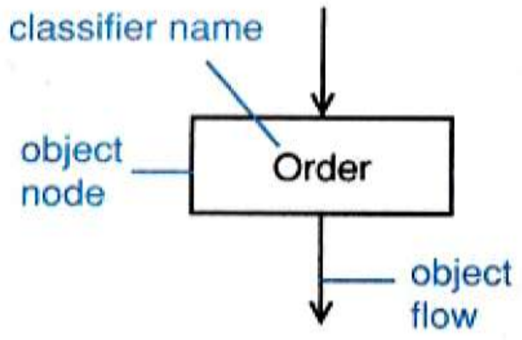


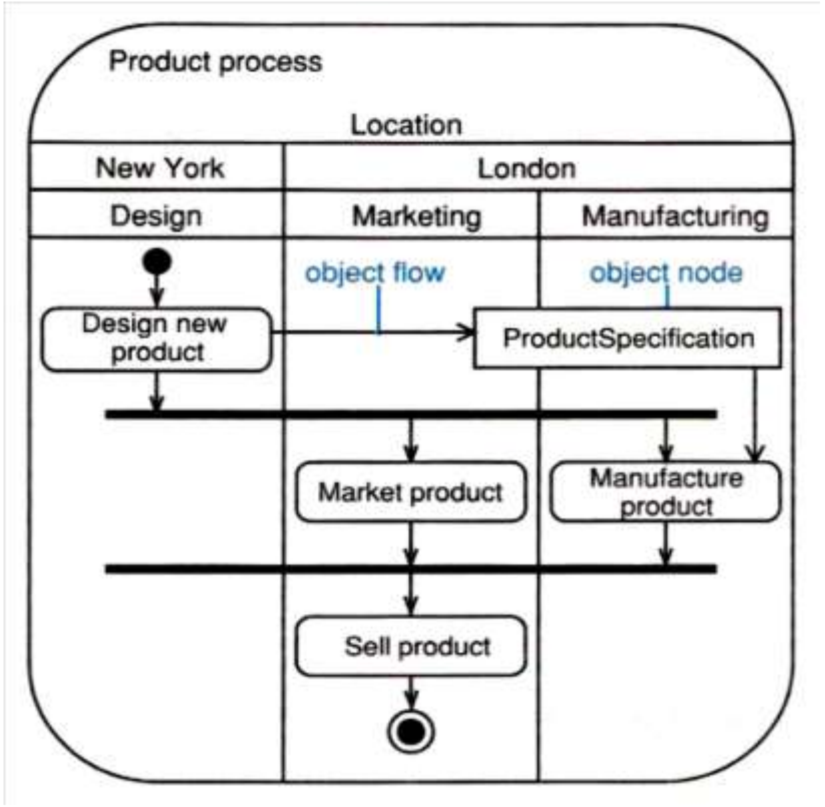
Control Nodes: Fork and Join



Object Nodes

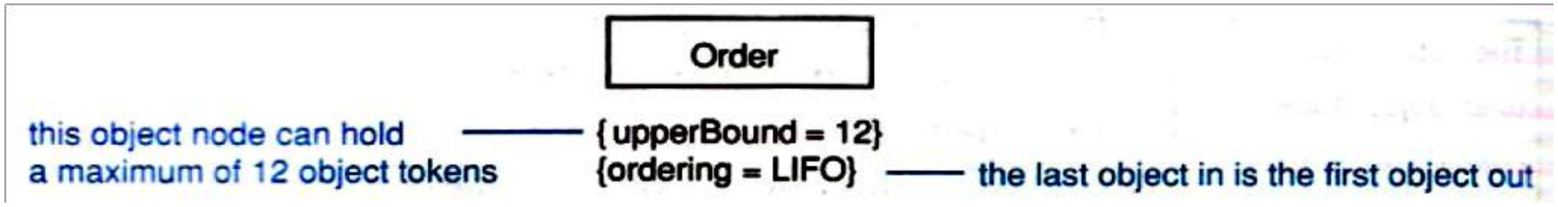
- Object nodes represent instances of a classifier.
- Input and output edges are object flows - represent the movement of objects.
- Object node output edges compete for each output token.

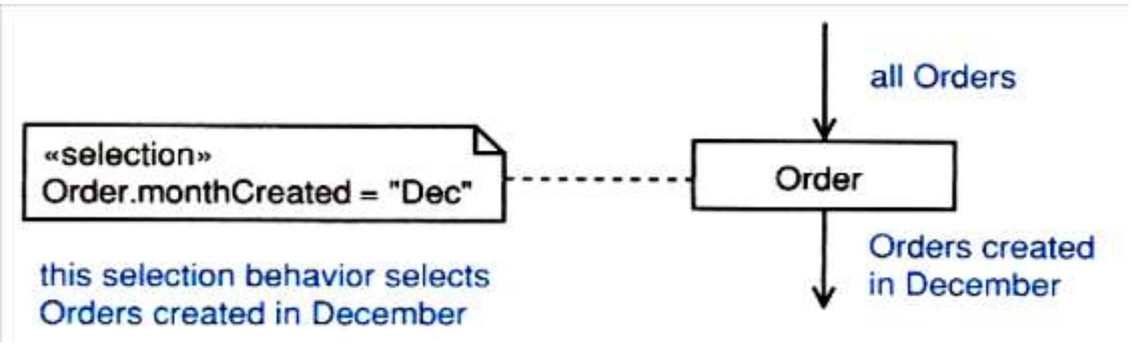


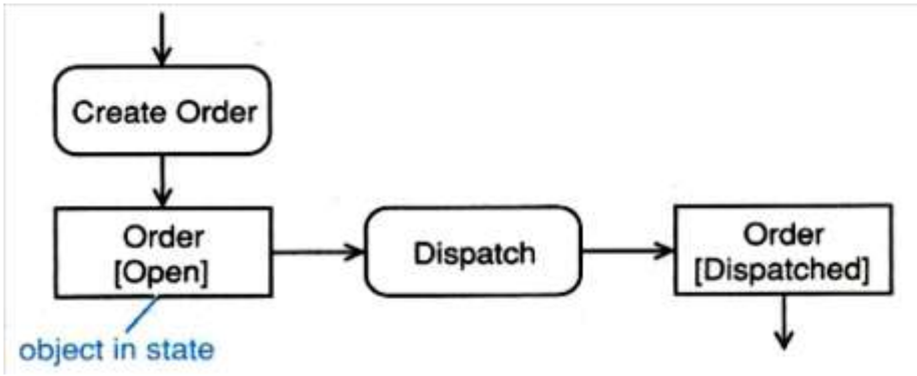


Object Nodes: Buffer Semantics

- Object nodes act as buffers:
  - {upperBound= n};
  - {ordering= FIFO} XOR {ordering= LIFO};
  - {ordering= FIFO} is the default;
  - may have a «selection».
- Object nodes can represent objects in a particular state.

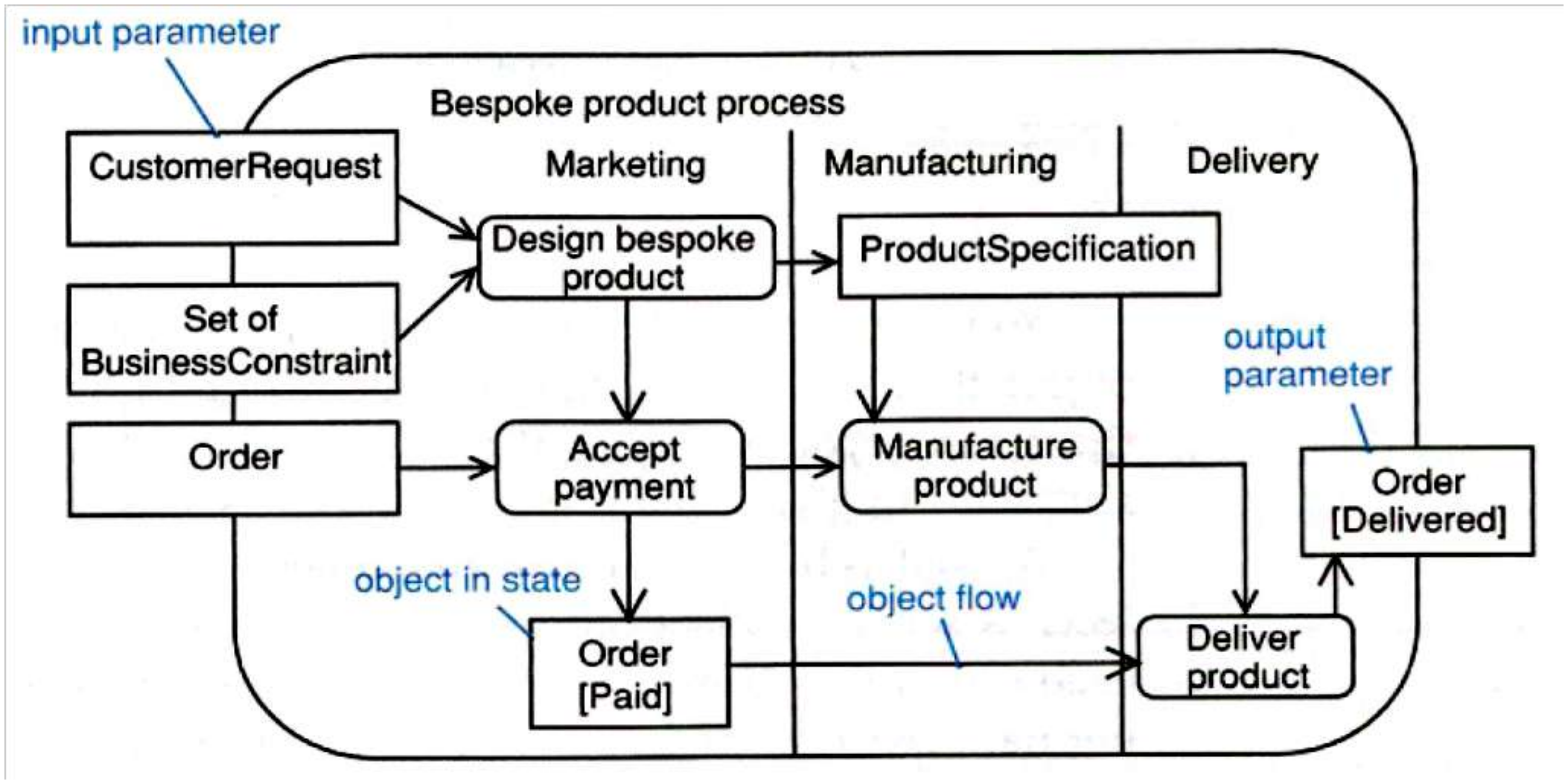

Order
this object node can hold a maximum of 12 object tokens
{upperBound = 12}
{ordering = LIFO}
the last object in is the first object out


all Orders
«selection» Order.monthCreated = "Dec"
Order
Orders created in December
this selection behavior selects Orders created in December



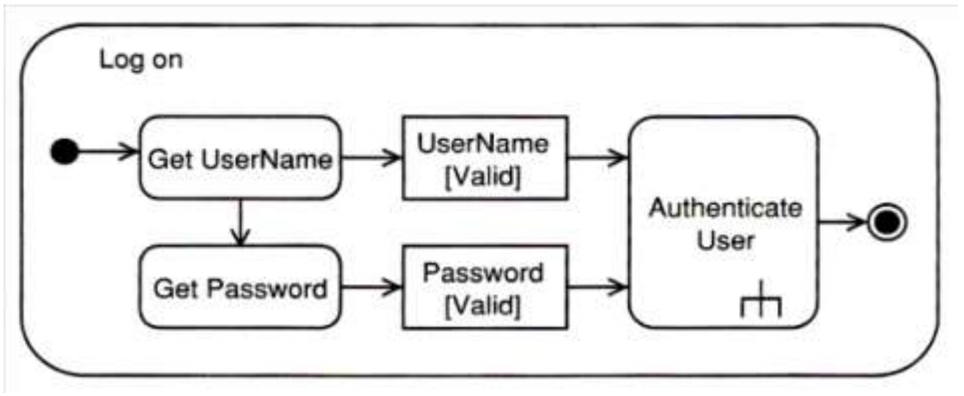
Object Nodes: Activity Parameters

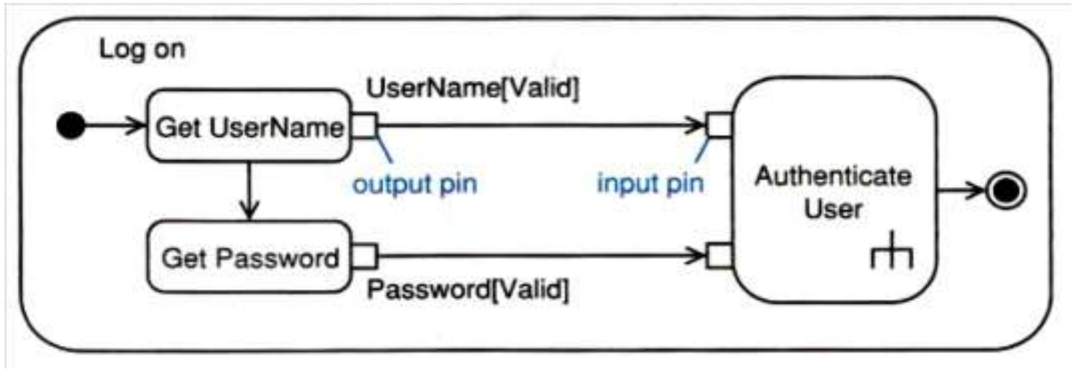
- Activity parameters are object nodes input to or output from an activity:
  - drawn overlapping the activity frame;
  - input parameters have one or more output edges *into* the activity;
  - output parameters have one or more input edges *out of* the activity.



Pins

- A Pin is an object node that represents one input to or output from an action or activity.





Reference

- Arlow, J., Neustadt, I., *UML 2 and the Unified Process: Practical Object-Oriented Analysis and Design*, 2nd Ed. Addison-Wesley, 2005.