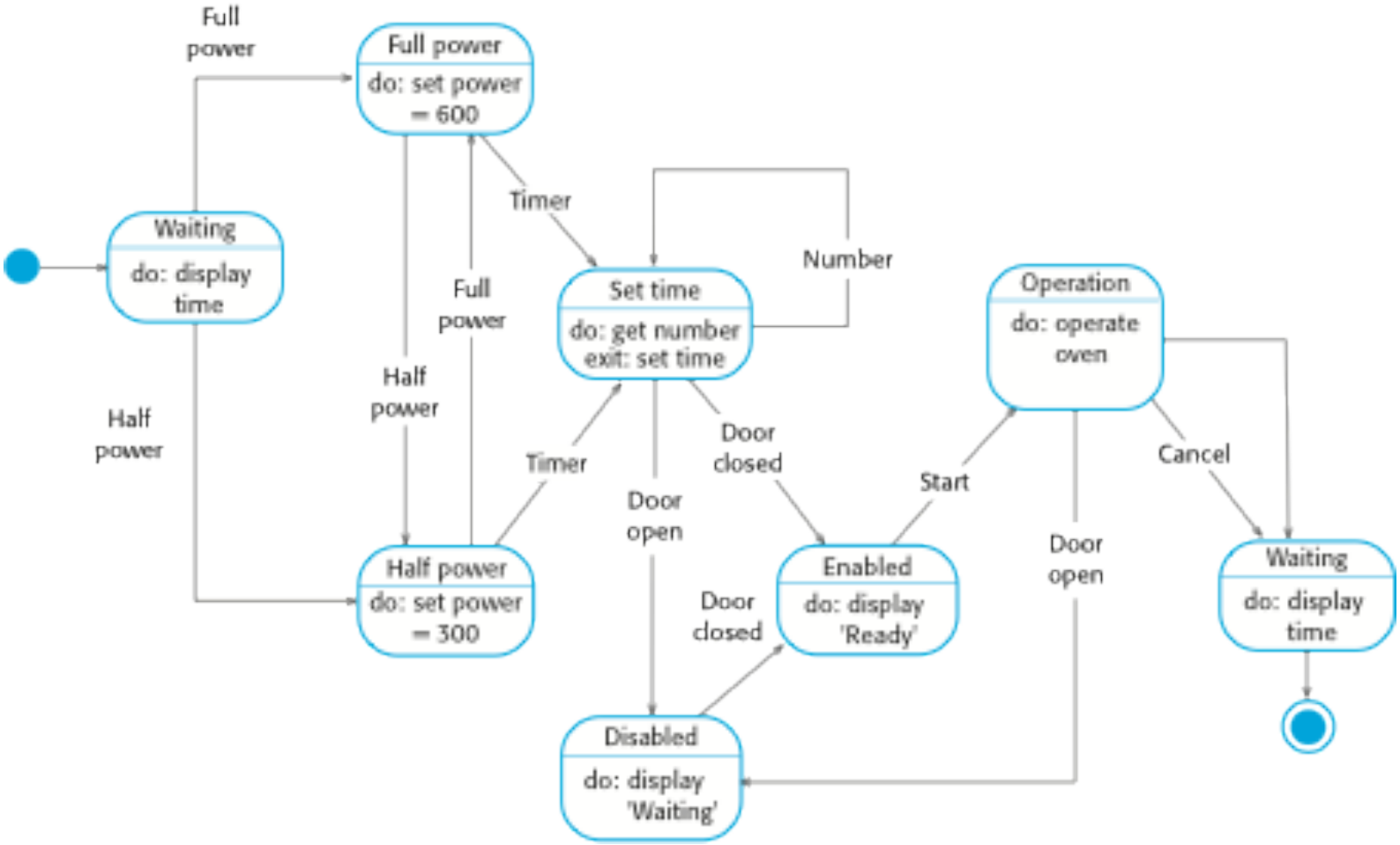
Modeling with UML

State Diagrams

- Model the behaviour of the system in response to external and internal events.
- Show the system's responses to stimuli so are often used for modelling real-time systems.
- Show system states as nodes and events as arcs between these nodes. When an event occurs, the system moves from one state to another.
- State diagrams are an integral part of the UML and are used to represent state machine models.

State diagram of a microwave oven



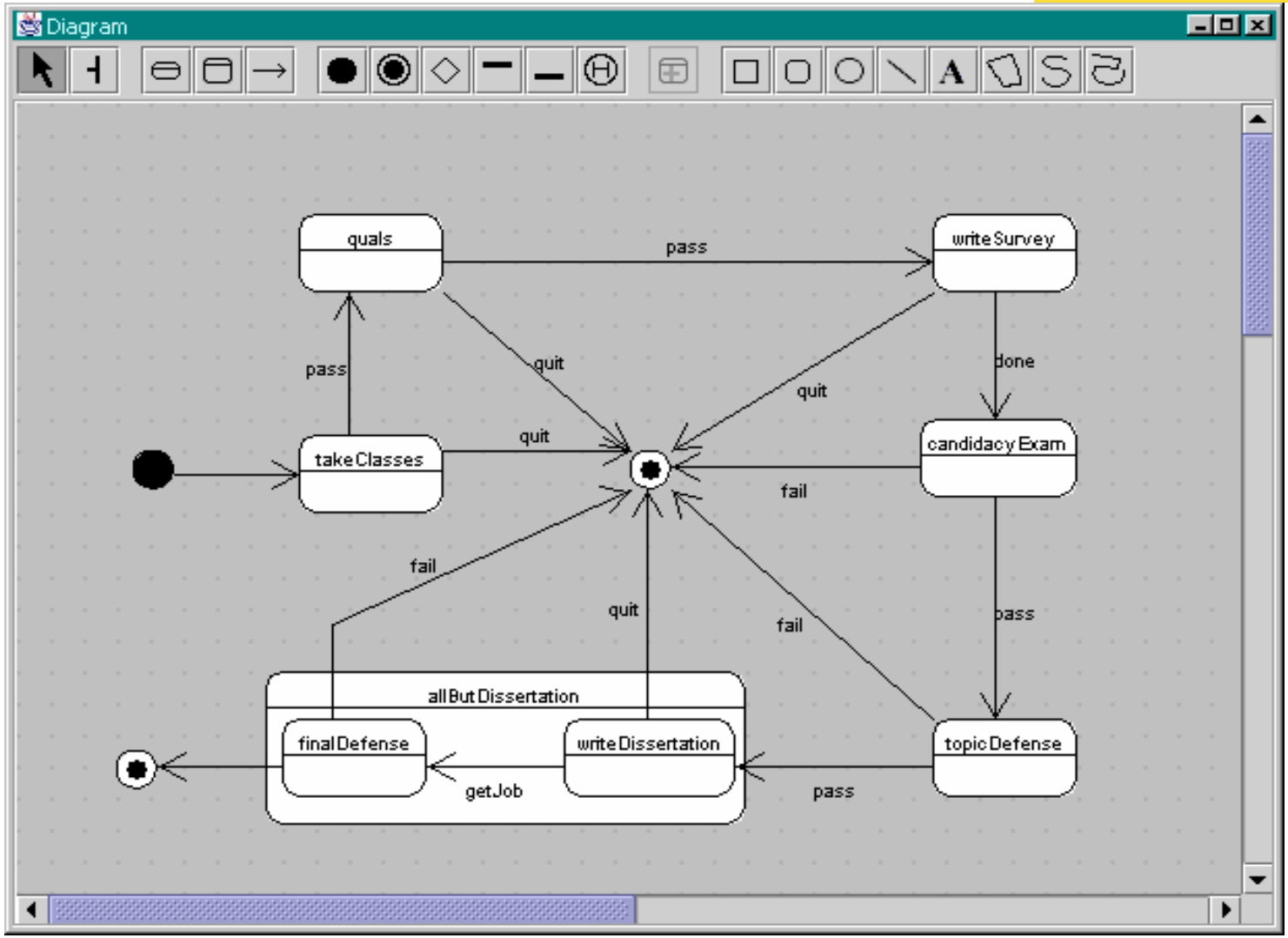
States & stimuli for the microwave oven

State	Description
Waiting	The oven is waiting for input. The display shows the current time.
Half power	The oven power is set to 300 watts. The display shows ‘Half power’.
Full power	The oven power is set to 600 watts. The display shows ‘Full power’.
Set time	The cooking time is set to the user’s input value. The display shows the cooking time selected and is updated as the time is set.
Disabled	Oven operation is disabled for safety. Interior oven light is on. Display shows ‘Not ready’.
Enabled	Oven operation is enabled. Interior oven light is off. Display shows ‘Ready to cook’.
Operation	Oven in operation. Interior oven light is on. Display shows the timer countdown. On completion of cooking, the buzzer is sounded for five seconds. Oven light is on. Display shows ‘Cooking complete’ while buzzer is sounding.

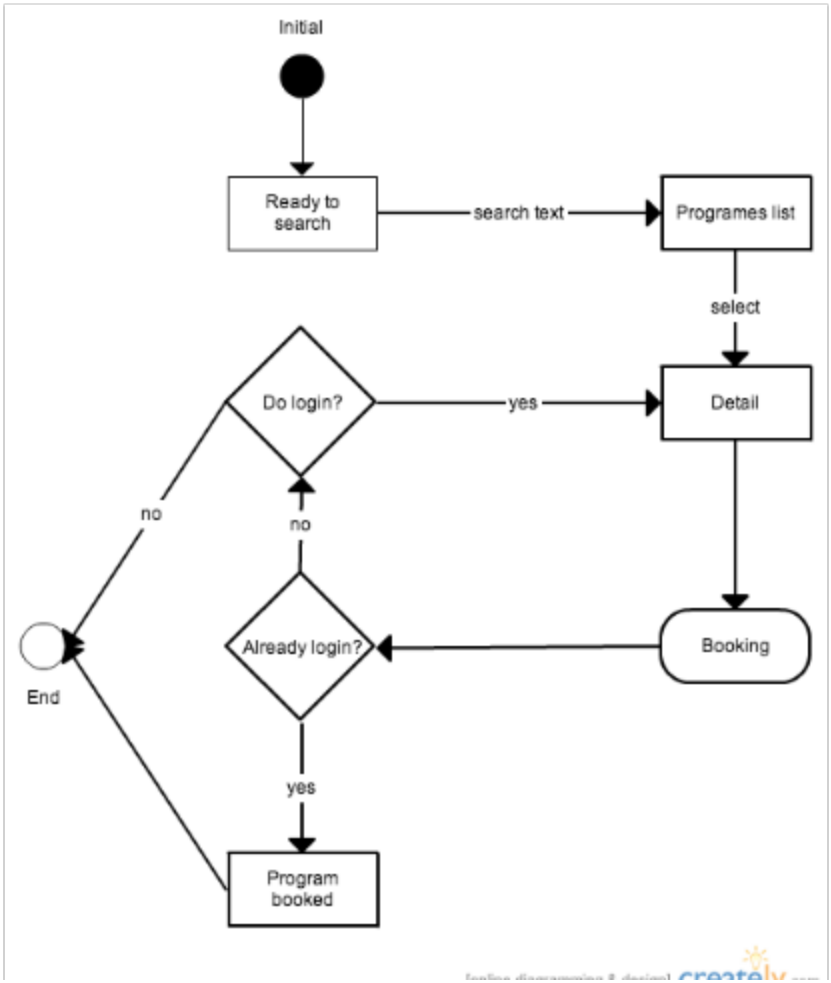
States & stimuli for the microwave oven

Stimulus	Description
Half power	The user has pressed the half-power button.
Full power	The user has pressed the full-power button.
Timer	The user has pressed one of the timer buttons.
Number	The user has pressed a numeric key.
Door open	The oven door switch is not closed.
Door closed	The oven door switch is closed.
Start	The user has pressed the Start button.
Cancel	The user has pressed the Cancel button.

State Diagram – PhD candidate



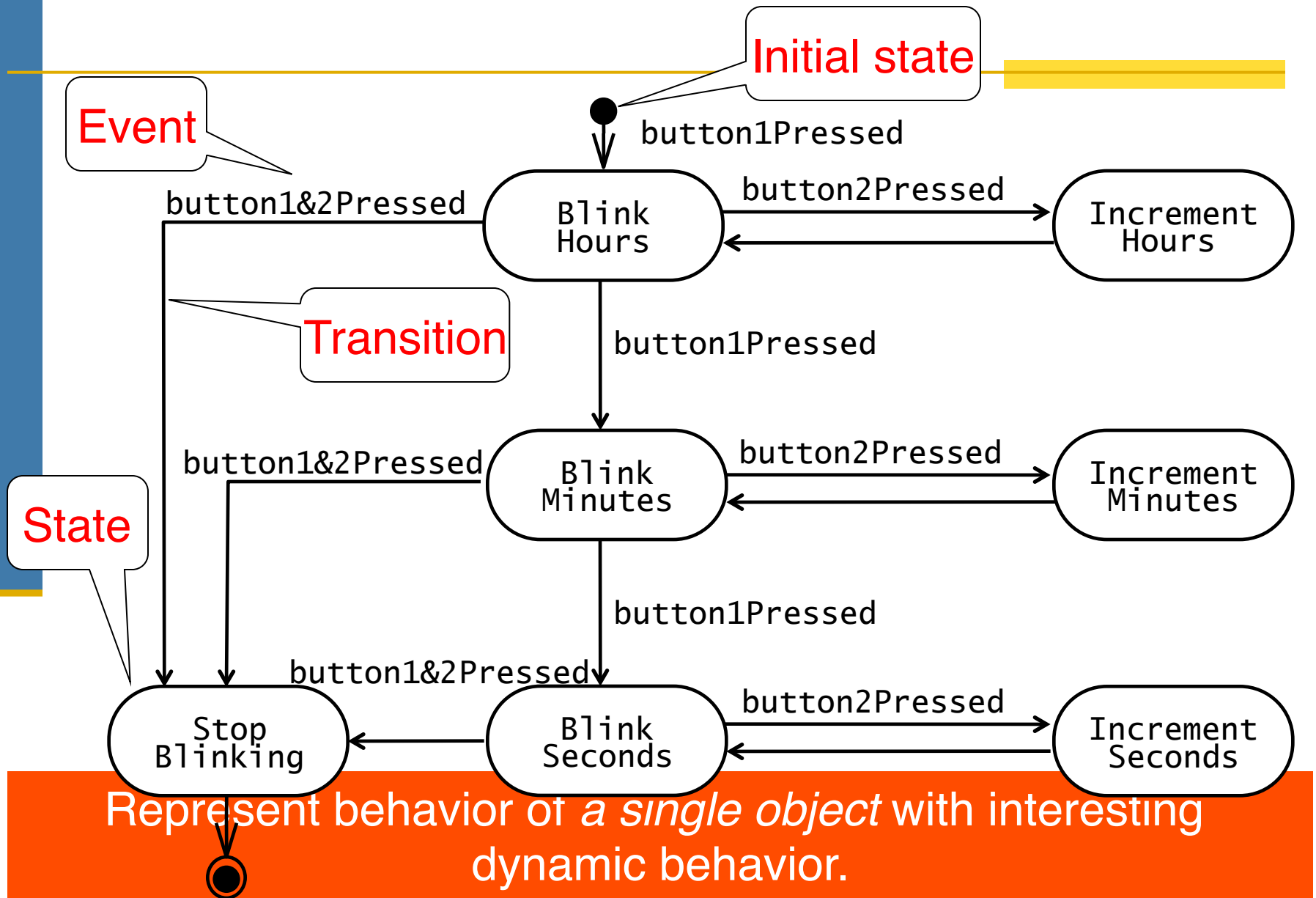
State Diagram – Seminar Search



Exercise: State Diagram

- Draw a state diagram that models the behavior of LCD display of the SimpleWatch.

Exercise: State diagram

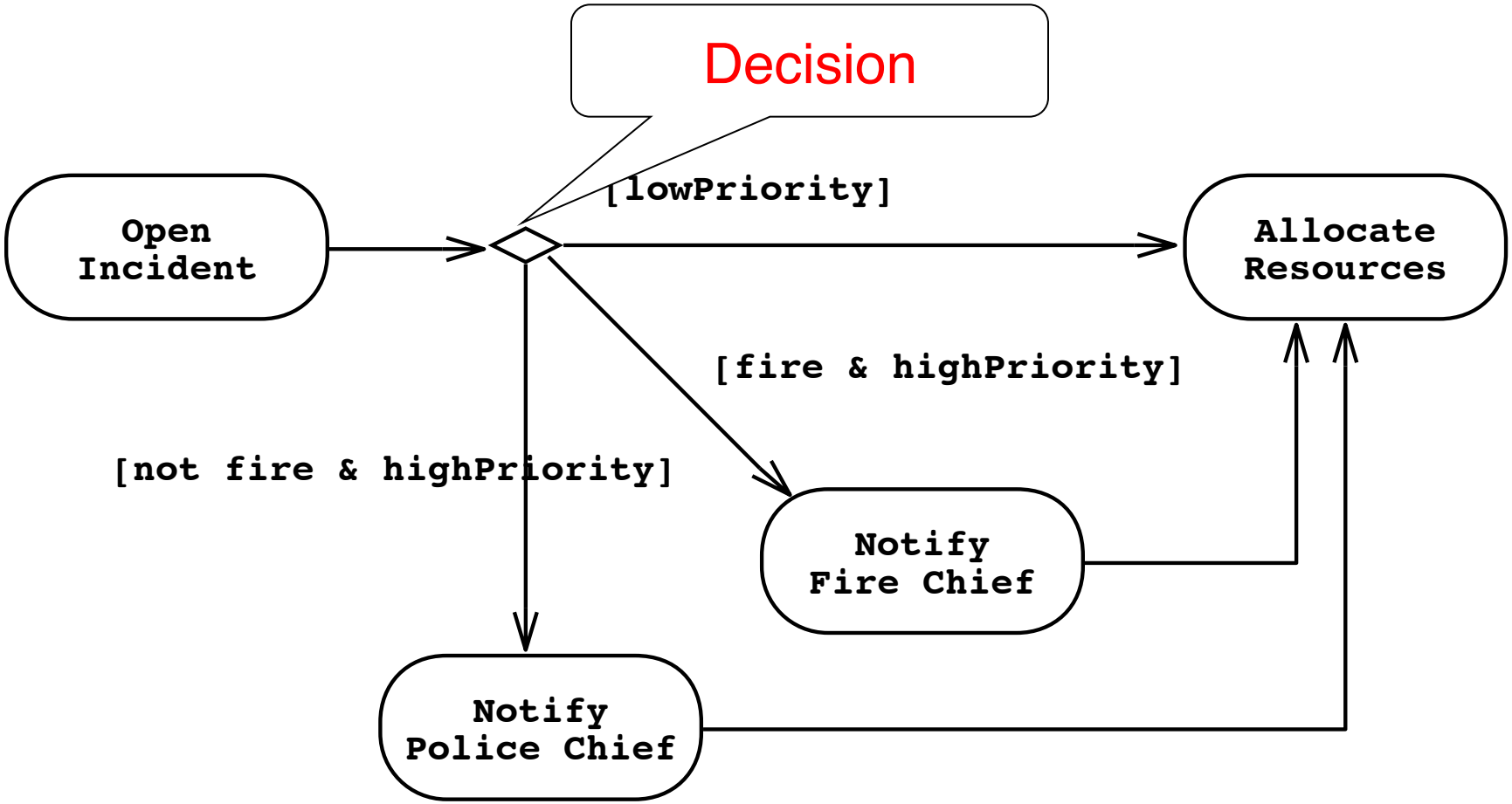


Represent behavior of *a single object* with interesting dynamic behavior.

Activity Diagrams

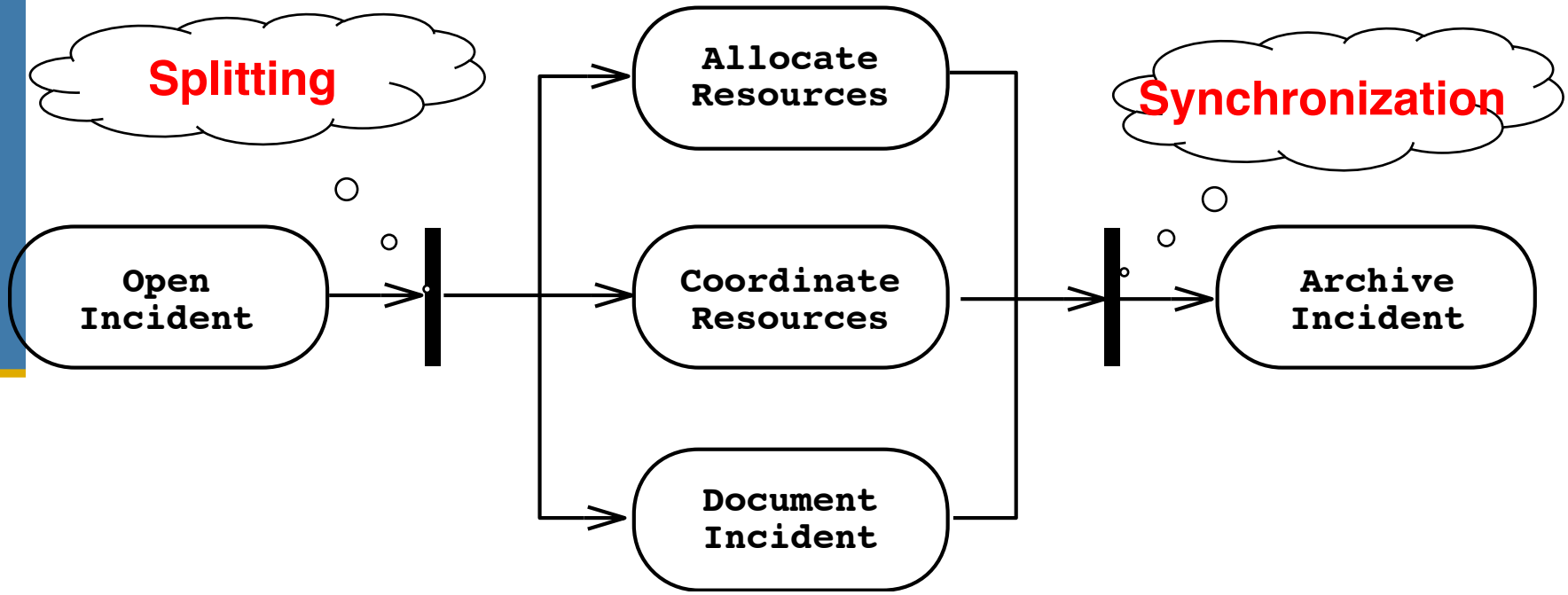
- An activity diagram is a special case of a state chart diagram
- The states are activities (“functions”)
- An activity diagram is useful to depict the workflow in a system

Activity Diagrams allow to model Decisions



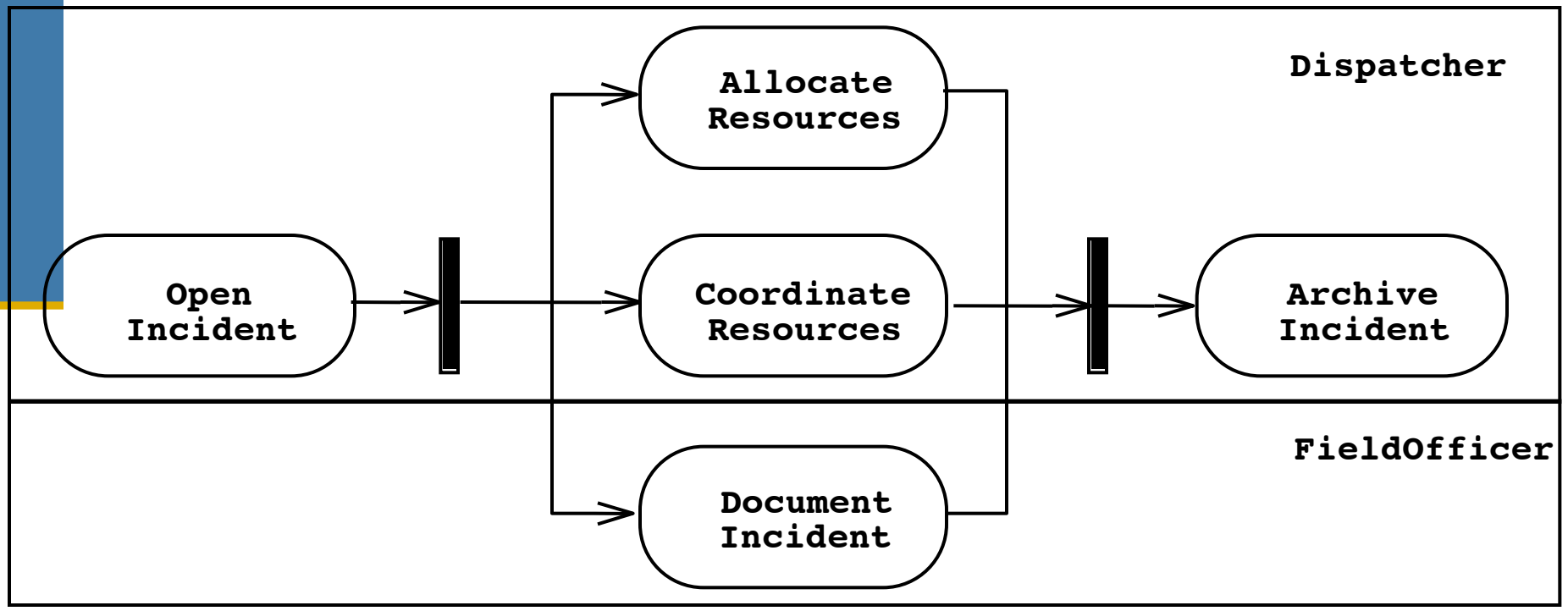
Activity Diagrams can model Concurrency

- Synchronization of multiple activities
- Splitting the flow of control into multiple threads



Activity Diagrams: Grouping of Activities

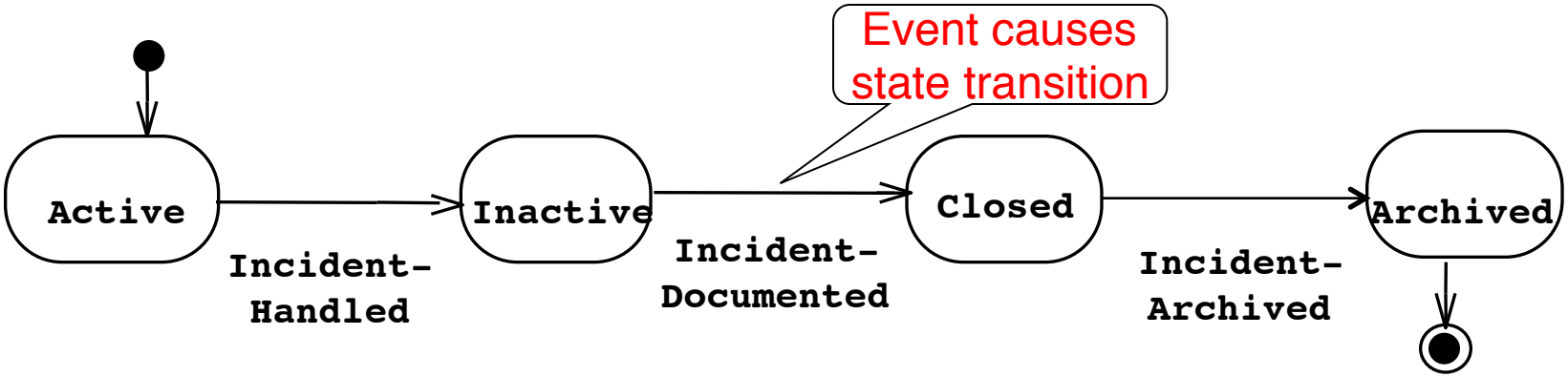
- Activities may be grouped into swimlanes to denote the object or subsystem that implements the activities.



Activity Diagram vs. Statechart Diagram

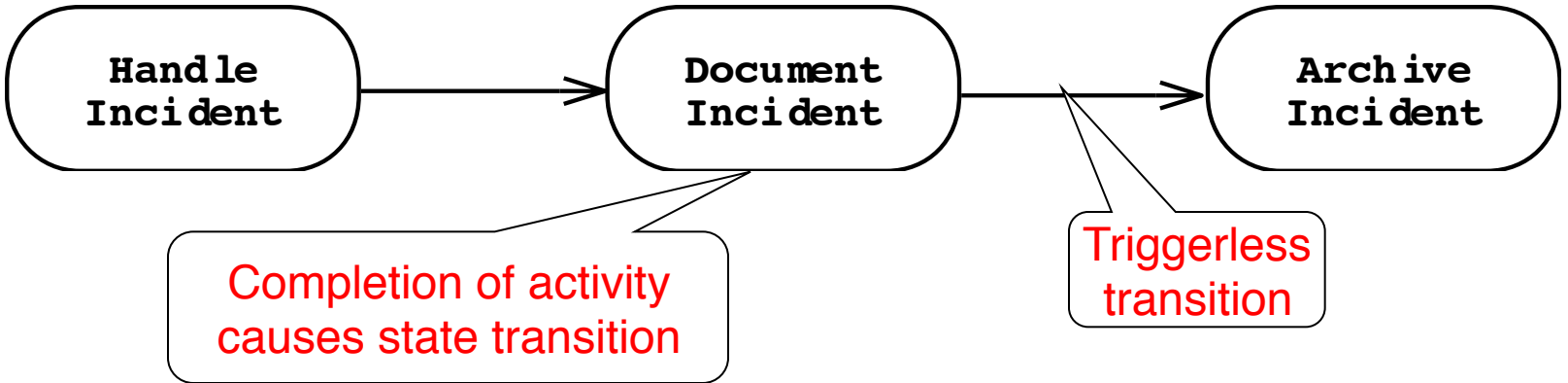
Statechart Diagram for Incident

Focus on the set of attributes of a single abstraction (object, system)

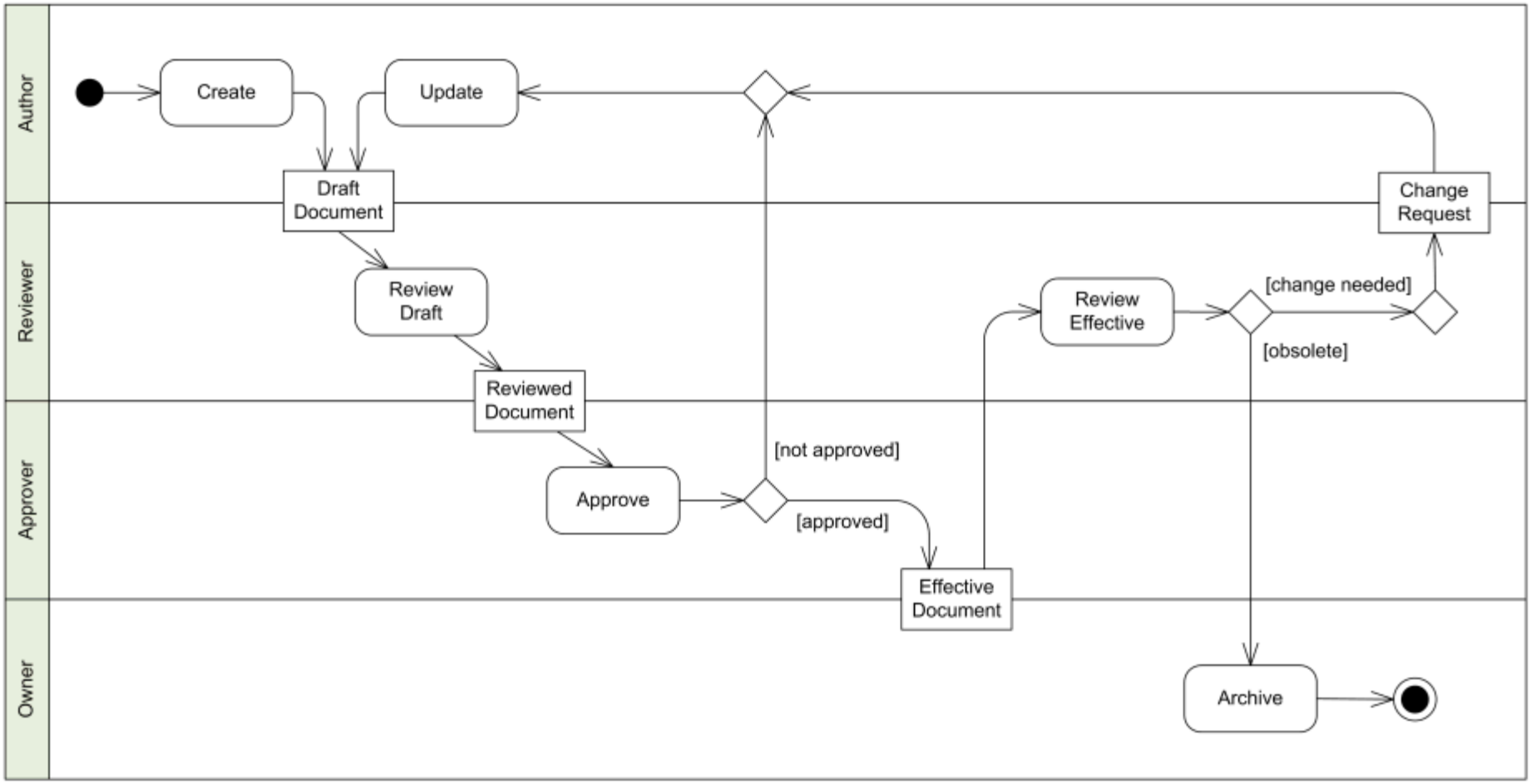


Activity Diagram for Incident

(Focus on dataflow in a system)



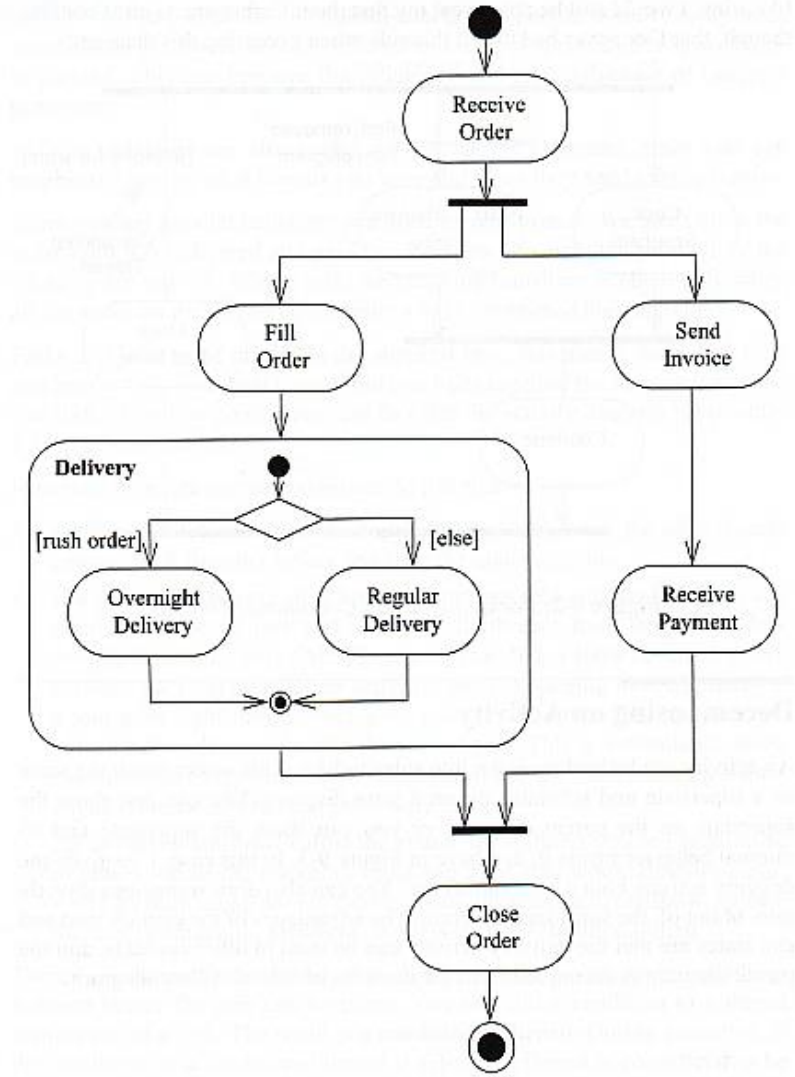
Activity Diagram – document management process



Exercise: Activity Diagram

- Draw an activity diagram for fulfillment of an order, (i.e., an order is received, it is processed, invoice is produced, delivered either overnight or regular, and payment is received before the order is closed).

Exercise: Activity Diagram



UML Summary

- UML provides a wide variety of notations for representing many aspects of software development
  - Powerful, but complex
- UML is a programming language
  - Can be misused to generate unreadable models
  - Can be misunderstood when using too many exotic features
- We concentrated on a few notations:
  - Functional model: Use case diagram
  - Object model: class diagram
  - Dynamic model: sequence diagrams, statechart and activity diagrams

Additional References

- Martin Fowler
  - UML Distilled: A Brief Guide to the Standard Object Modeling Language, 3rd ed., Addison-Wesley, 2003
- Grady Booch,James Rumbaugh,Ivar Jacobson
  - The Unified Modeling Language User Guide, Addison Wesley, 2nd edition, 2005
- Commercial UML tools
  - Rational Rose XDE for Java
    - http://www-03.ibm.com/software/products/en/ratirosefami
  - Together (Eclipse, MS Visual Studio, JBuilder)
    - http://www.borland.com/Products/Requirements-Management/Together
- Open Source UML tools
  - http://java-source.net/open-source/uml-modeling
  - ArgoUML,UMLet,Violet, …