Object Oriented Analysis and Design with Java

Dr. Kamatchi Priya L
Department of Computer Science and Engineering

Acknowledgements: Significant portions of the information in the slide sets presented through the course in the class, are extracted from the prescribed text books, information from the Internet. Since these are only intended for presentation for teaching within PESU, there was no explicit permission solicited. We would like to sincerely thank and acknowledge that the credit/rights remain with the original authors/creators only

Object Oriented Analysis and Design with Java

Behaviour Modelling:
UML State Machine Diagrams and Models

Dr. Kamatchi Priya L

Department of Computer Science and Engineering

Behavoiur Modeling

- A behavior Modeling is intended to provide clarity, for example, about internal processes, business processes or the interaction of different systems.
- UML Behavioral Diagrams depict the elements of a system that are dependent on time and that convey the dynamic concepts of the system and how they relate to each other.
- The elements in these diagrams resemble the verbs in a natural language and the relationships that connect them typically convey the passage of time.

State Diagram

- State machine diagram typically are used to describe state-dependent behavior for an object.
- An object responds differently to the same event depending on what state it is in.
- The state model is a reductionist view of behaviour that examines each object individually.

State Diagram Vs Flowchart

- A flowchart illustrates processes that are executed in the system that change the state of objects.

- A state diagram shows the actual changes in state, not the processes or commands that created those changes.

How to Draw State Diagram

- Before you begin your drawing find the initial and final state of the object in question.
- Next, think of the states the object might undergo. For example, in e-commerce a product will have a release or available date, a sold out state, a restocked state, placed in cart state, a saved on wish list state, a purchased state, and so on.
- Certain transitions will not be applicable when an object is in a particular state, for example a product can be in a purchased state or a saved in cart state if its previous state is sold out.

State Diagram: Notations and Symbols

States

States represent situations during the life of an object. A state is represented by using a rectangle with rounded corners.

A simple state

A state with internal activities

State Diagram: Notations and Symbols

Transition

A solid arrow represents the path between different states of an object. Label the transition with the event that triggered it and the action that results from it. A state can have a transition that points back to itself.

State Diagram: Notations and Symbols

Initial State

A filled circle followed by an arrow represents the object's initial state.

State Diagram: Notations and Symbols

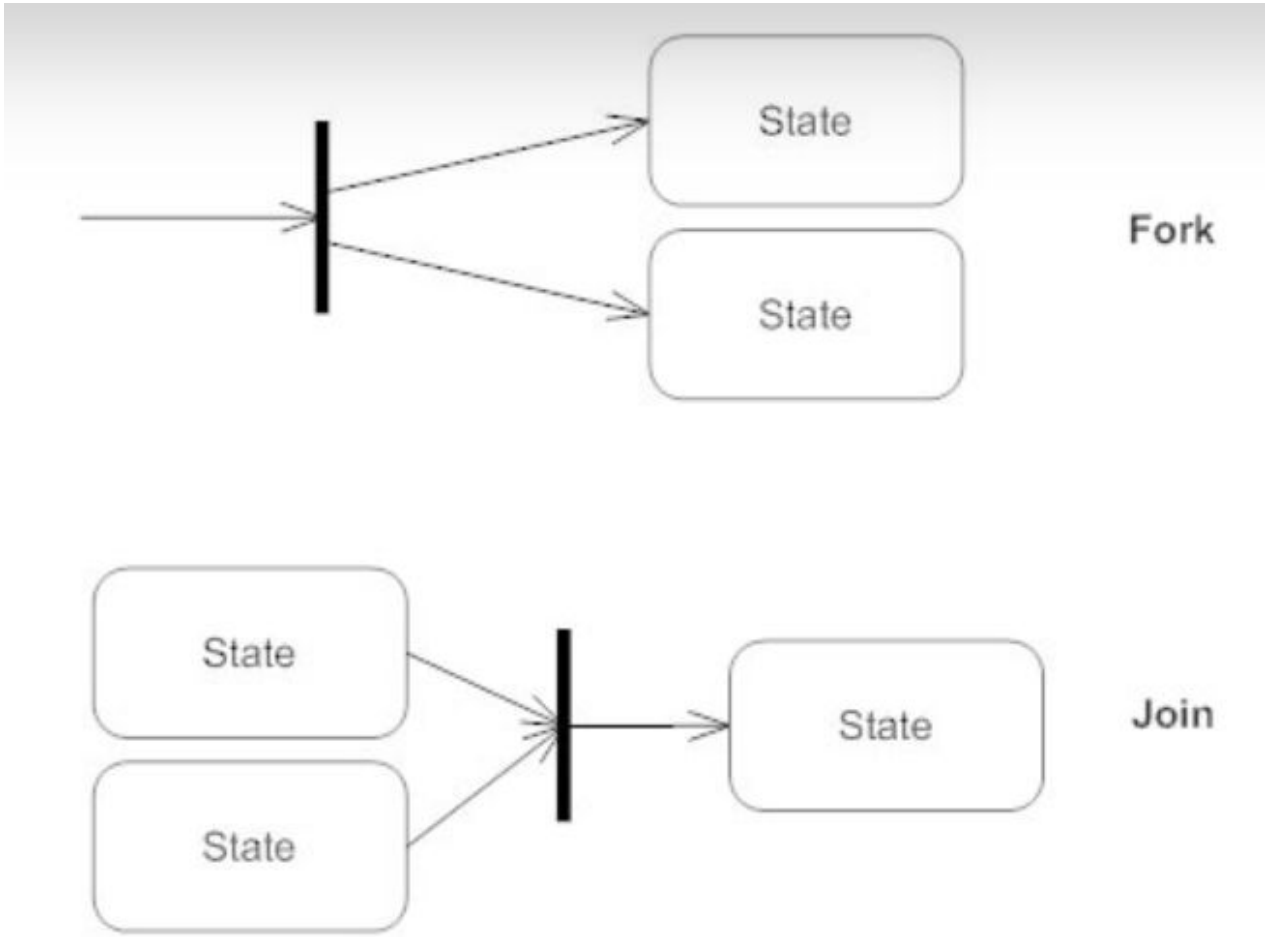
Final State

An arrow pointing to a filled circle nested inside another circle represents the object's final state

State Diagram: Notations and Symbols

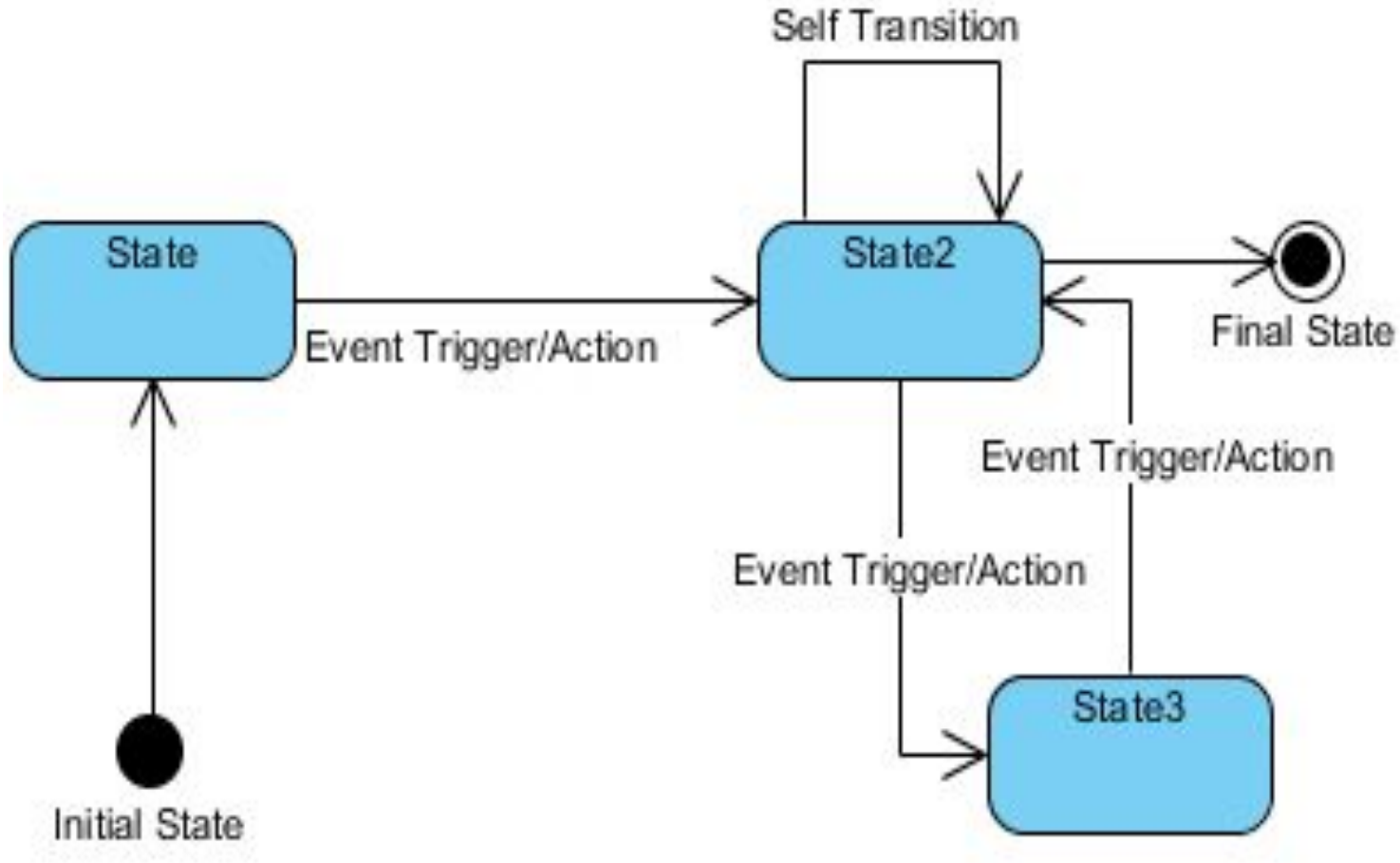
Synchronization and Splitting of Control

- A short heavy bar with two transitions entering it represents a synchronization of control.
- The first bar is often called a fork where a single transition splits into concurrent multiple transitions.
- The second bar is called a join, where the concurrent transitions reduce back to one.



Triggers Event, Action and Guard Condiotion

- A trigger is an **event** that initiates a transition from one state to another.
- A **guard condition** is a Boolean condition that must be satisfied for a transition to occur.
- An effect is the **action** or activity that happens when a transition occurs.



Adding triggers to state machine diagrams

- **Adding call events to state machine diagrams**

  A call event is an event that represents the receipt of a request to invoke an operation. A transition with a call event initiates when the called operation is invoked.

- **Adding change events to state machine diagrams**

  A change event is an event that represents a condition. The condition is defined by a Boolean-valued expression that triggers a transition when its value changes from false to true.

- **Adding signal events to state machine diagrams**

  A signal event represents a specific message that initiates a transition when an object receives it.

Adding triggers to state machine diagrams

- **Adding time events to state machine diagrams**

  A time event is an event that represents the passage of a defined period of time or an absolute time. A transition with a time event trigger initiates when the time value is satisfied.

- **Adding any received event triggers to state machine diagrams**

  Add triggers to state machine diagrams for any received event, which means that any type can trigger the event. This type of transition triggers on any event, as long as no other event trigger type is connected to the source state.

- **Adding call events to state machine diagrams**

  A call event is an event that represents the receipt of a request to invoke an operation. A transition with a call event initiates when the called operation is invoked.

Adding triggers to state machine diagrams

- **Adding change events to state machine diagrams**

  A change event is an event that represents a condition. The condition is defined by a Boolean-valued expression that triggers a transition when its value changes from false to true.

- **Adding signal events to state machine diagrams**

  A signal event represents a specific message that initiates a transition when an object receives it.

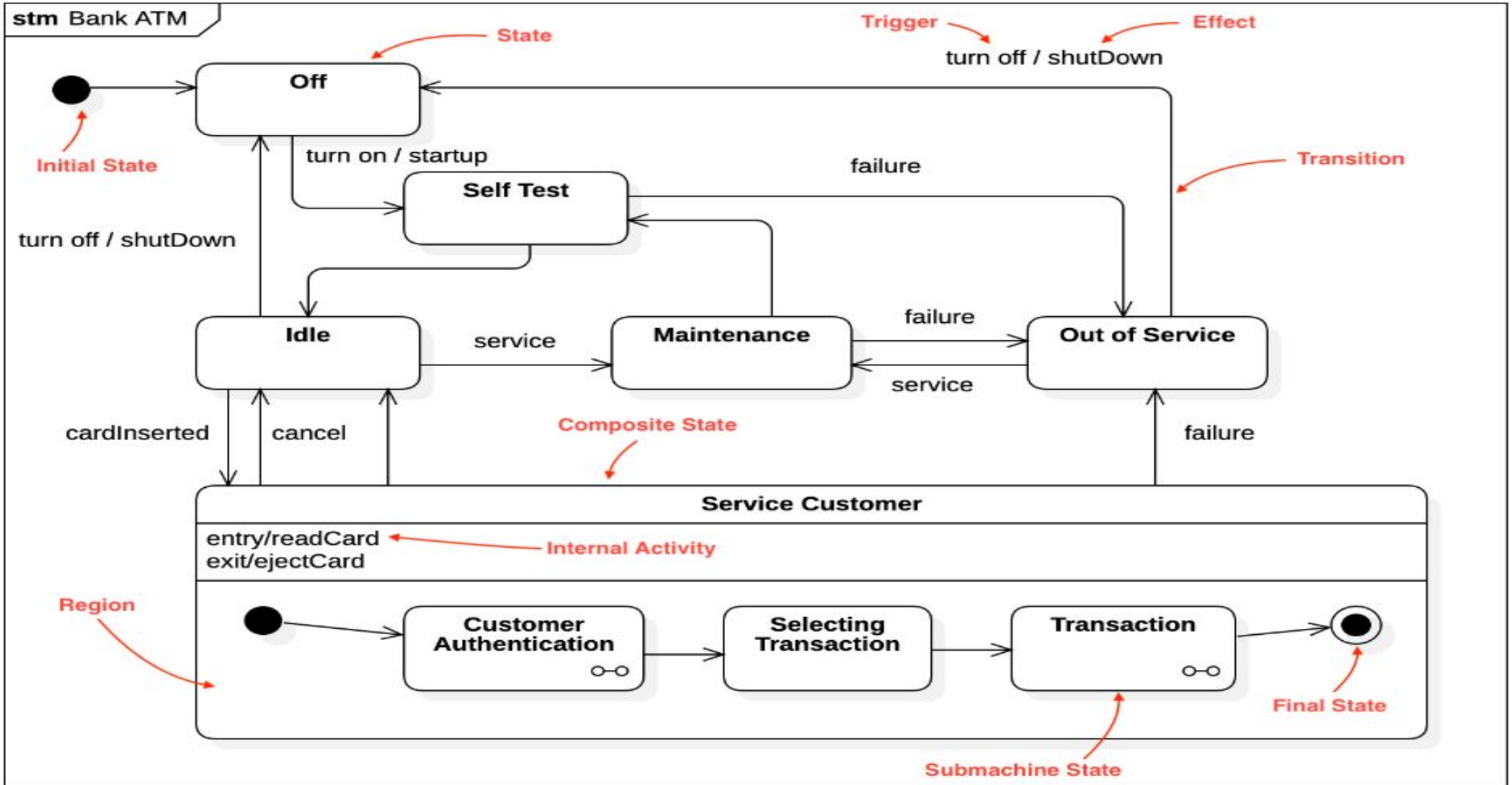
- **Adding time events to state machine diagrams**

  A time event is an event that represents the passage of a defined period of time or an absolute time. A transition with a time event trigger initiates when the time value is satisfied.

- **Adding any received event triggers to state machine diagrams**

  Add triggers to state machine diagrams for any received event, which means that any type can trigger the event. This type of transition triggers on any event, as long as no other event trigger type is connected to the source state

Bank ATM


stm Bank ATM
State
Trigger
Effect
turn off / shutDown
Off
Initial State
turn on / startup
failure
Transition
Self Test
turn off / shutDown
Idle
service
Maintenance
failure
Out of Service
service
cardInserted
cancel
Composite State
failure
Service Customer
entry/readCard
exit/ejectCard
Internal Activity
Region
Customer Authentication
Selecting Transaction
Transaction
Final State
Submachine State

THANK YOU

Dr. Kamatchi Priya L

Department of Computer Science and Engineering

priyal@pes.edu