

Topics - State Modeling



- Events
- States
- Transitions and
- Conditions
- State Diagrams

- **Changes to the objects and their relationships over time.**
- Those aspects of a system with **time and changes are called state model.**
- **Control is the aspect** of the system that describes the **sequences of operations** that occur in response to external stimuli

- The major state modeling concepts are:
 - **Events** :- represent external stimuli
 - **States** :- represent values of objects
 - **State diagrams** :- graphical representation of finite state machine

1.Events

- An **event** is an **occurrence** at a **point in time**.
- One **event** may **logically precede or follow another**, or the two events may be unrelated.
- **Two events** that are causally **unrelated or related**
- **Causally unrelated event** is said to be **concurrent**; they have no effect on each other. Concurrent events can occur in any order.
 - Two events are causally **related**. E.g. Flight 123 must depart Chicago after it can arrive in San Francisco.
 - The two events are causally **unrelated**. E.g. Flight 123 may depart before or after Flight 456 departs Rome.

1.Events contd.

- An event is a **one way transmission of information from one object to another.**
- Events include **error conditions** as well as **normal occurrences.**
- Eg :-transaction aborted , time out
- Note : **often appear past tense verb (ex: power turned on) or on set condition (temperature becomes lower than freezing)**

Kinds of Events



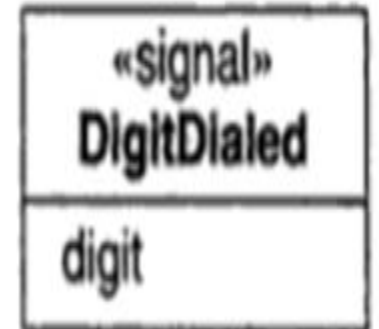
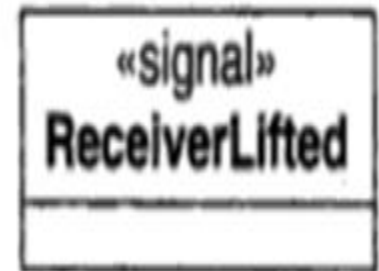
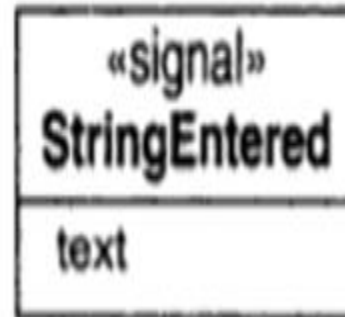
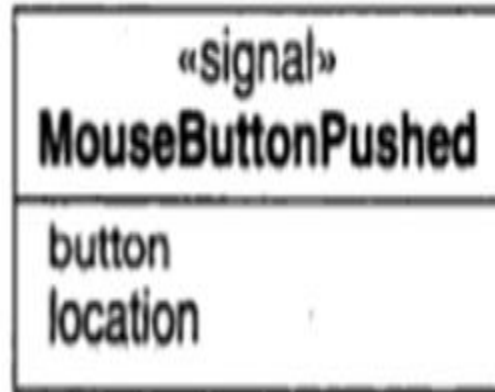
There are several kinds of events, the most common are

- Signal event
- Change event
- Time event

Signal Event

- A **signal** is an **explicit one-way transmission** of information from one object to another
- A **signal event** is the event of sending or receiving a signal.
- Event is **not like a subroutine call that returns a value**
 - An object sending an event to another may expect a reply, but the reply is a separate event under the control of the second object, which may or may not choose to send it.
 - Note: A signal is a message between objects but a signal event is an occurrence in time

Signal Class and Attributes



Signal classes and attributes. A signal is an explicit one-way transmission of information from one object to another.

Change Events



- A change is an event that is **caused by the satisfaction of a Boolean expression.**
- The uses of change event is tested continuously: whenever the expression changes from false to true
- The UML notation : **when** keyword followed by a **parenthesized** Boolean expression
- Example of change events
 - **When** (room temperature < heating set point)
 - **When** (room temperature > cooling set point)

Time Events

- A time event is an event **caused by the occurrence of an absolute time or the elapse of a time interval.**
- The UML notation : The notation for a time interval is the keyword **when** / **after** followed by a parenthesized expression
- Example of time events
 - **When** (date= January 1, 2014)
 - **After** (10 seconds)

2.States

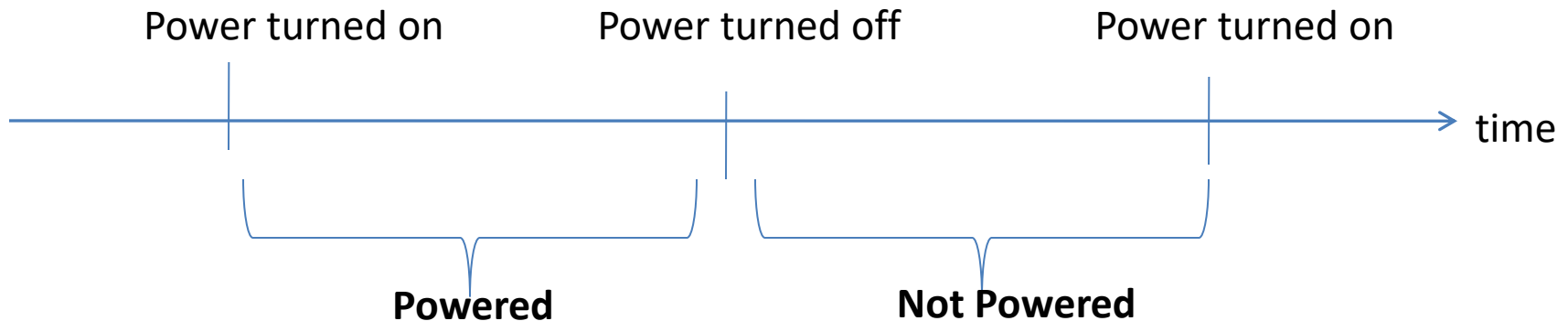
- It is **often associated with a continuous activity**. Ex. Telephone ringing
- A STATE is an **abstraction of the attribute values and links of an object**
- It **has duration** (occupies an interval of time)
- A **state** is drawn as a **rounded box** containing an optional name

State - Examples

Waiting

Dialing

Powered



States

1. Represent **values** of object
2. The attribute values and links held by an object
3. A state **has duration**. it occupies an interval of time
4. State **separates** two **events**

Events

1. Represent **external stimuli**
2. An individual stimulus from one object to another
3. An event **has no duration**. It happens at a point of time
4. An event **separates** two **states**

3.Transitions



- A **change of state caused by an event** is called **transitions**.
- It is an **instantaneous change** from one state to another.
- The source and target of a transition are **different states or same state**.
- The choice of next state depends on both the original state and the event received.
- The UML notation for a transition is a **line with arrowhead** from the source state to the target state
- Example of transition:
 - When a called phone is **answered**, the phone line transitions from the **Ringing** state to the **Connected** state.

4. Conditions

- A **guard condition** is a **Boolean expression** that must be true in order for a transition to occur.
- A state can be defined in terms of a condition that **valid over interval of time**
- Conditions can be used as **guards on transitions**
- **A guards transitions fires** when its event occurs , but only if the **guard condition is true**
- a **guard condition** is checked only once
- A **condition** is listed within **square brackets []** after an event name
- For example
 - a person goes out in the morning (event),
 - if the temperature is below freezing (condition) , then put on your gloves(next state).

5.State Diagrams



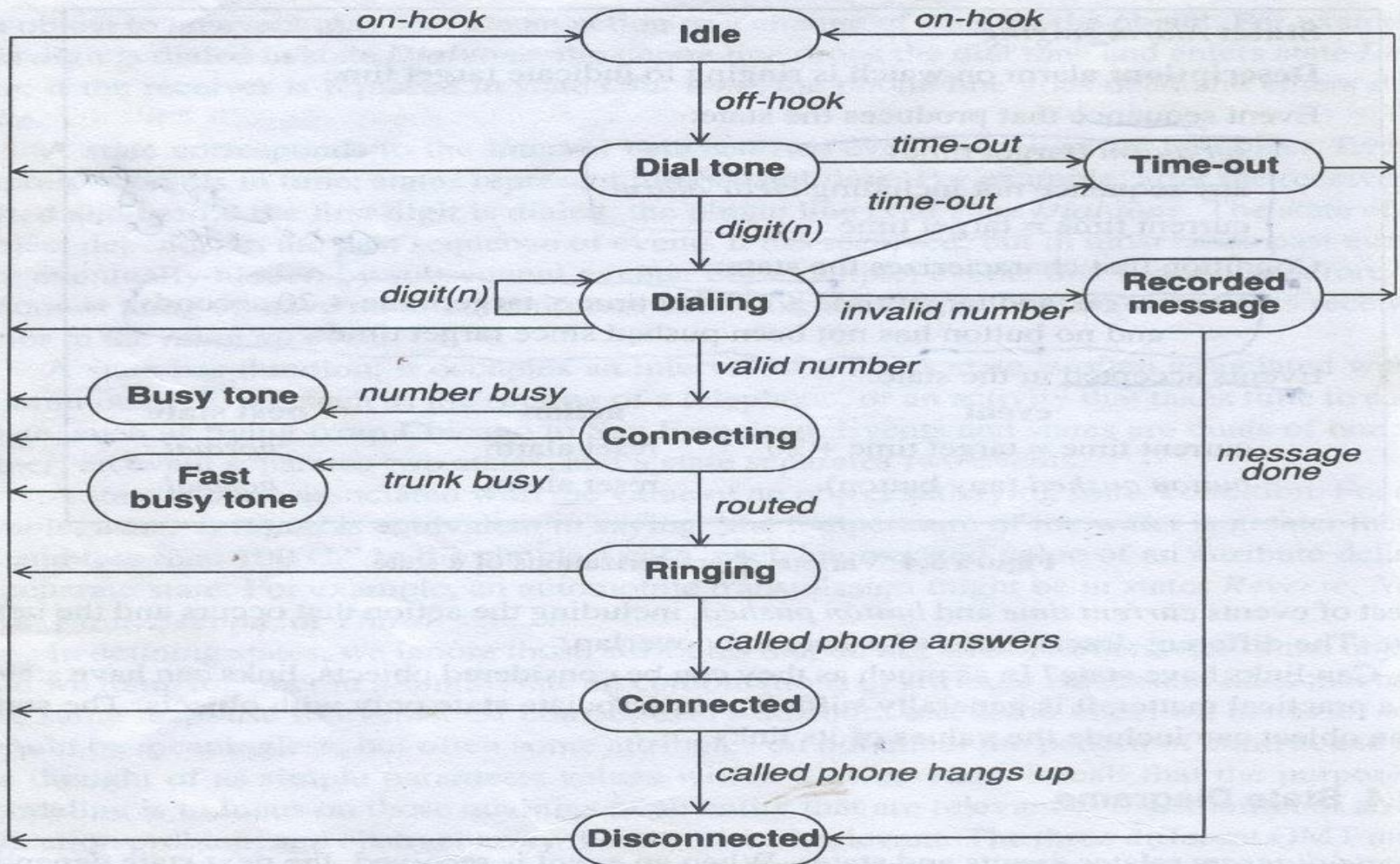
- A **state diagram** is a **graph** whose **nodes** are **states** and whose **directed arcs** are **transitions** labeled by an event names
- A state diagram specifies the **state sequences caused by event sequences.**
- State names must be unique within the scope of a state diagram
- **Note:** state diagrams can implement by direct implementation or by converting into programming code

5.State Diagrams



- The state model consists of multiple state diagrams, one state diagram for each class with important temporal behavior
- State diagram contains sequences associated with normal call as well as abnormal call (time out while dialing , getting busy line)
- States can represent
 - continuous loops
 - one shot life cycles

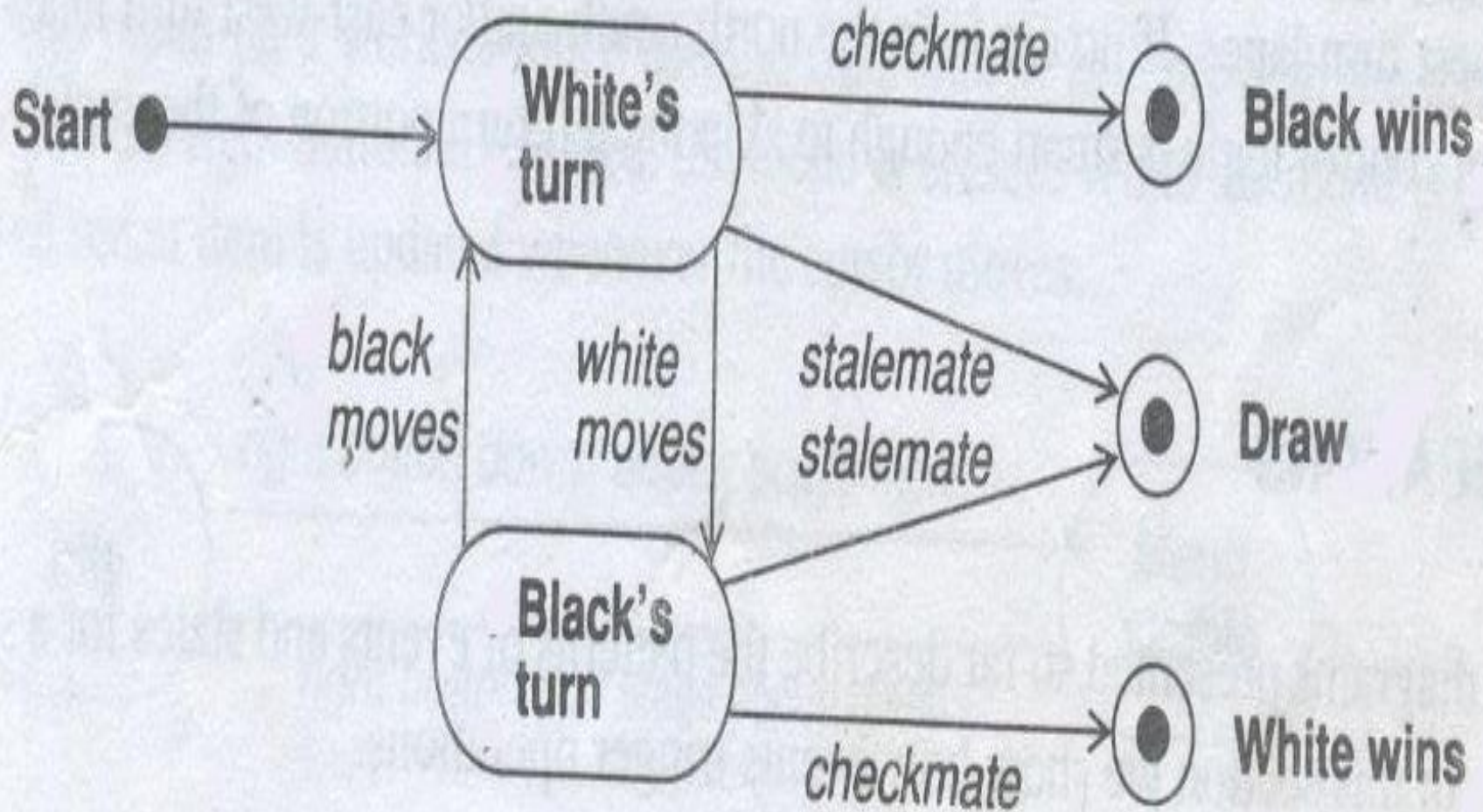
State diagram(continuous loop) for phone line



One-shot life cycles

- It represents objects with finite lives
- It has initial and final states
- The **initial state** is entered on **creation of an object**
- An **initial state** is shown by a **solid circle**
- The circle can be labeled to indicate different initial conditions
- **Final state** implies **destruction of the object**
- The **final state** shown by a **bull's eye**. It distinguished from initial state

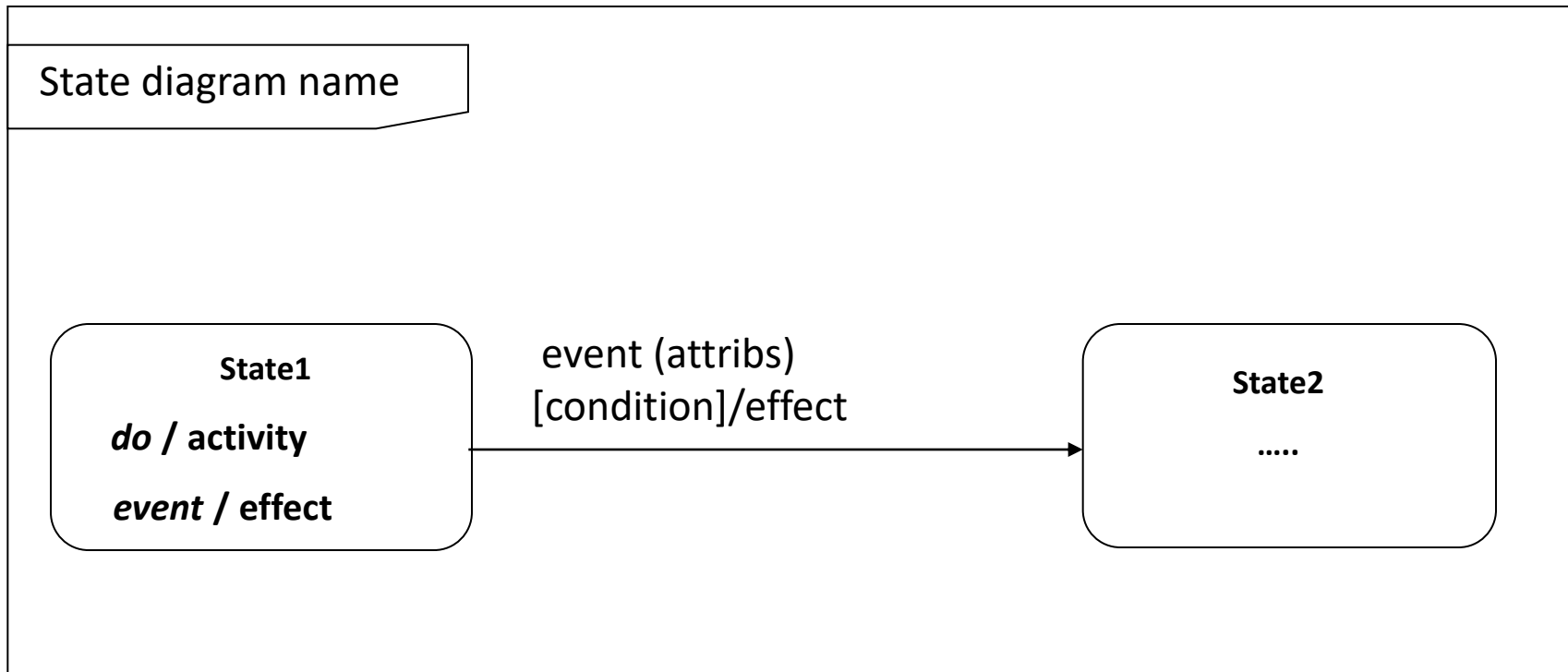
State diagram(One-shot life cycles) for chess game



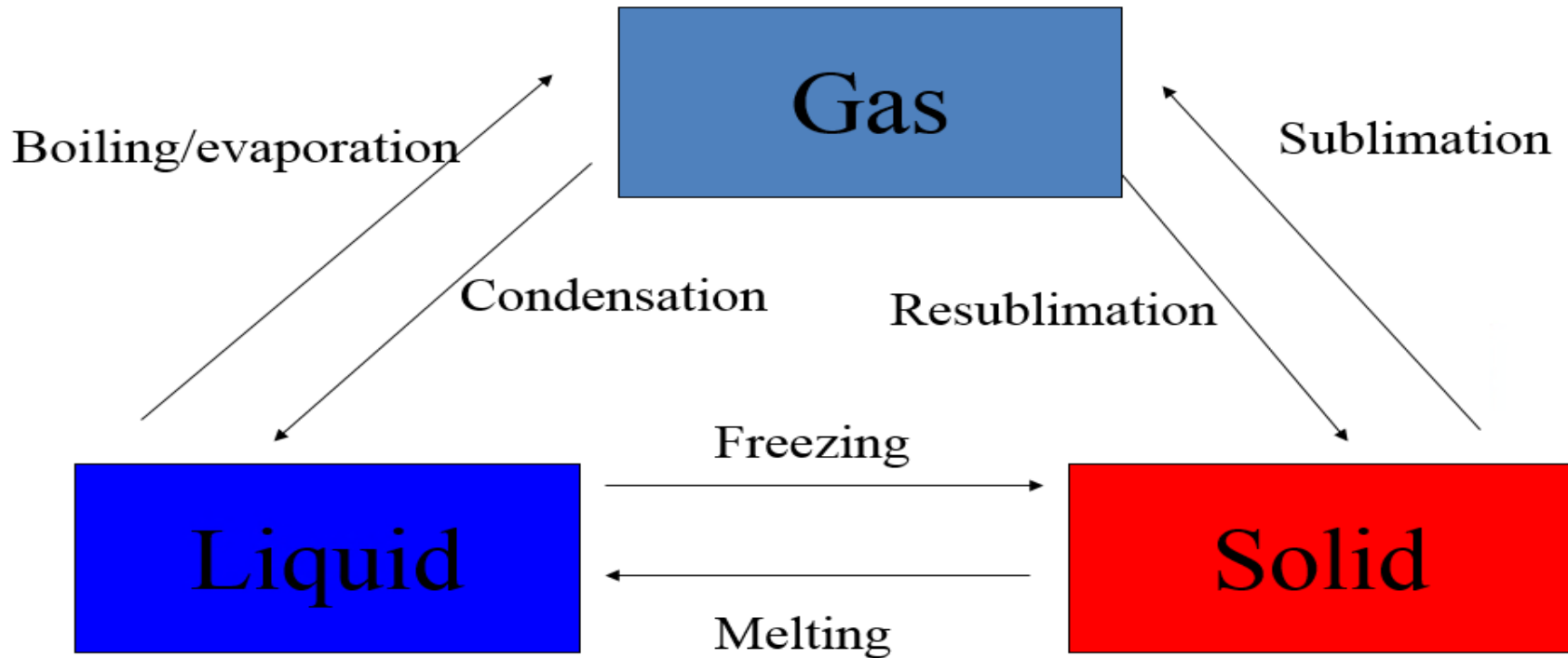
Summary of Basic State Diagram Notation

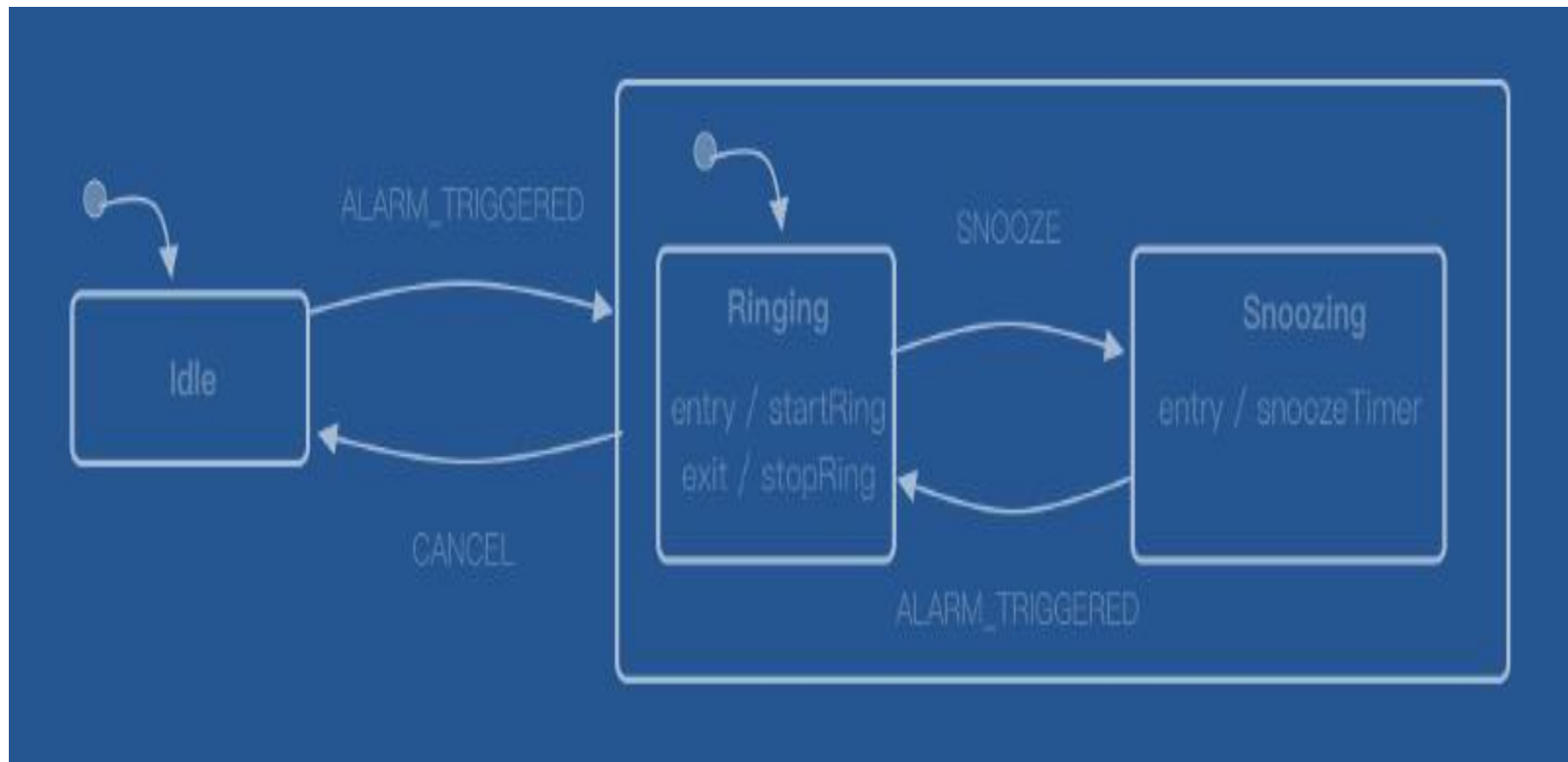


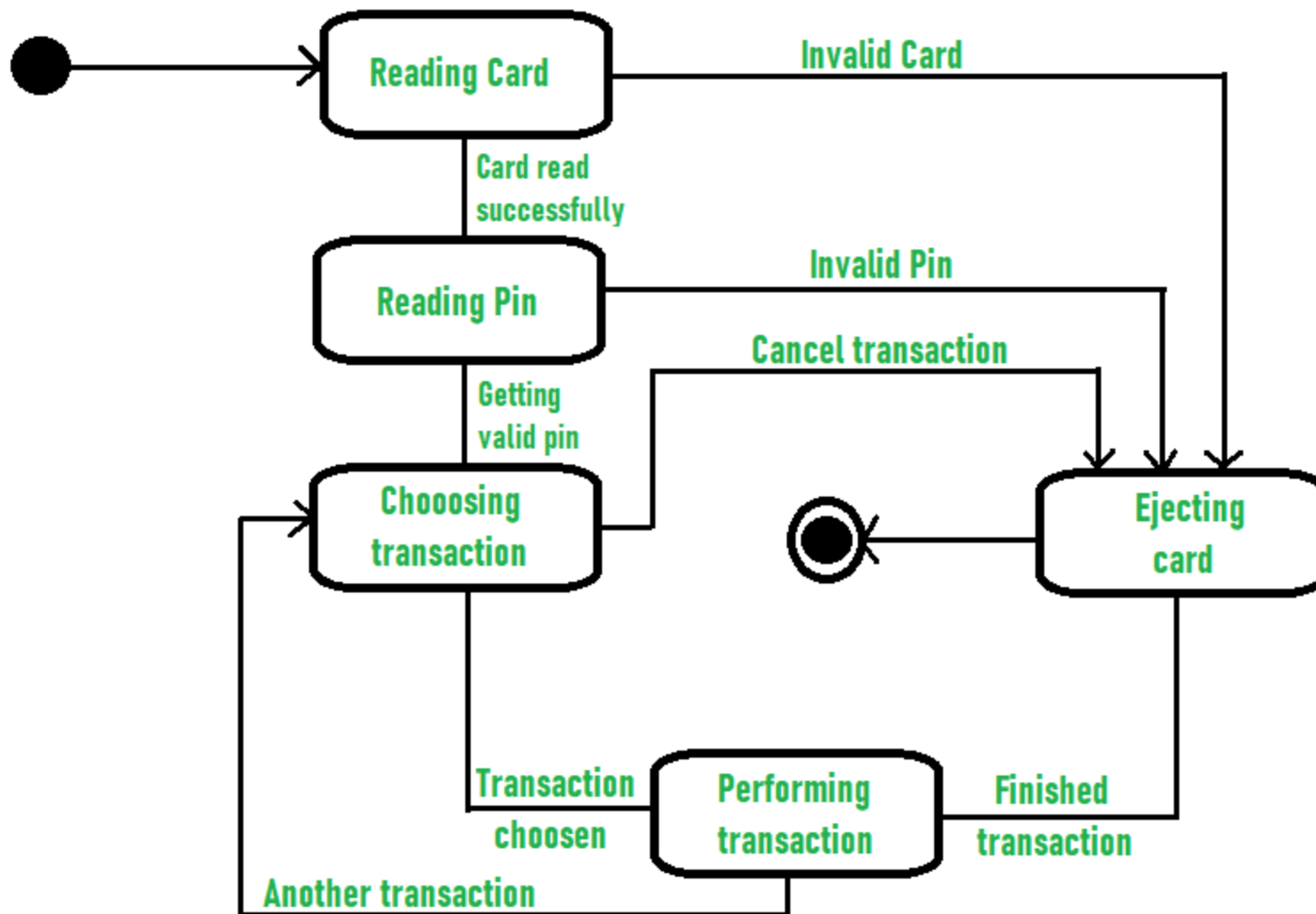
- The following Figure summarizes the basic UML syntax for state diagrams.



Change of state







State Transition Diagram for ATM System