State Diagrams & Advanced Activity Diagrams

Software Design and Analysis CSCI 2040

Objectives

- Introduce UML State Machine Diagram Notation.
- Create State Machine Diagrams for Classes and Use Cases.
- Advanced techniques in Activity Diagrams.

State Diagrams Notation

State Machine Diagrams

"Usability is like oxygen— you never notice it until it is missing"

Introduction

- As with Activity Diagrams, UML State
 Diagrams show a dynamic view.
- The UML includes notation to illustrate the events and states of:
 - things, transactions, use cases, people, and so forth.

State Diagram: Telephone



Notation

- State Diagram illustrates the interesting events and states of an object, and the behavior of an object in reaction to an event.
 - Transitions are shown as arrows, labeled with their event.
 - States are shown in rounded rectangles.
 - It is common to include an initial pseudo-state,
 - which automatically transitions to another state when the instance is created.

Definitions: Events, State and Transitions

Event - is a significant or noteworthy occurrence.

- e.g., A telephone receiver is taken off the hook.
- State is the condition of an object at a moment in time between events.
 - e.g., A telephone is in the state of being "idle" after the receiver is placed on the hook and until it is taken off the hook.

Transition - when an event occurs, the object moves from the prior state to the subsequent state.

When the event "off hook" occurs, transition the telephone from the "idle" to "active" state.

Lifecycle of Object

- A state machine diagram shows the lifecycle of an object:
 - what events it experiences
 - its transitions
 - and the states it is in between these events
- It does not need to illustrate every possible event;
 - if an event arises that is not represented in the diagram, the event is ignored as far as the state machine diagram is concerned

Modeling Changing States in Domain Model

How to model changing states in Domain Model?

Modeling Changing States in Domain Model

Note that a payment does not stay in one of these states; it typically transitions from unauthorized to authorized.



Subject of State Diagram

- A state diagram may be applied to a variety of UML elements, including:
 - classes (conceptual or software)
 - use cases

Design Model State Diagram

- In UP, any element in any model (Domain Model, Design Model, and so forth) may have a State Diagram
 - to better understand or communicate its dynamic behavior in response to events.
- For example, a statechart associated with the Sale design class of the Design Model is itself part of the Design Model.

Use Case State Diagram

- A useful application of state diagrams is to describe the legal sequence of external system events that are handled by a system in the context of a use case.
 - For example:
 - During the Process Sale use case in the NextGen POS application, it is not legal to perform the makeCreditPayment operation until the endSale event has happened.

Process Sale

It illustrates that it is not legal to generate a *makePayment* event if an *endSale* event has not previously caused the system to transition to the *WaitingForPayment* state.



Utility of Use Case State Diagrams

- In a complex domain with many system events, the conciseness of use case statechart diagrams help a designer ensure that nothing is missed.
 - For example, the system should not be allowed to receive a payment unless a sale is complete;
 - code must be written to guarantee that.

Use Case State Diagram for POS Application



State Independent and Dependent Objects

- If, for all events of interest, an object always reacts the same way, it is a state-independent object.
- By contrast, state-dependent objects react differently to events depending on their state.
 - process control and telecommunication domains often have many state-dependent objects
 - e.g., a telephone is very state-dependent. Its response to pushing buttons depends whether it is on-hook or off-hook and whether there is an ongoing call or not.

Additional State Diagram Notation

- The UML notation for statechart diagrams contains a rich set of features.
 - transition actions
 - guard conditions
 - nested states

Transition Action and Guard Notation



Nested States

- A state allows nesting to contain substates
- Substates may be graphically shown by nesting them in a superstate box.



Nested States

Substates inherit the transitions of the superstate.

 When a transition to the Active state occurs, creation and transition into the PlayingDialTone substate occurs.



- Some User Interface applications, especially Web UI applications, have complex page flows.
- State machines are a great way to document that, for understanding, and a great way to model page flows, during creative design.
 - It is useful to model the flow between these pages.

Web Page Navigation Modeling



Complex Reactive Objects

Physical Devices – controlled software

- Phone, Car, Microwave oven
 - Reaction depends upon its current state
- Transactions and Business Objects
 - Sale, Order, Payment
 - What should happen to an *Order* if a cancel event occurs
- Role Mutators these are objects that change their role
 - A person changing roles from being a civilian to veteran.
 - Each role is represented by a state.

Advanced Activity Diagram Notation

Advanced Activity Diagrams

"If it wasn't backed-up, then it wasn't important."

- The Sysadmin Motto

Extending Notation

- Basic UML activity diagram includes action, fork, join, and object node.
- This can be extended, e.g., by incorporating partitioning.
 - Partitions show different parties involved in the process.

UML Activity Diagram



Course Registration System

The readability and usability of Activity Diagrams is better.



More UML Activity Diagram Notation

- How to show that an activity is expanded in another activity diagram?
 - Illustrate, using the rake symbol.

Expansion of Activity



Signals

There's more UML activity notation available. For instance, signals



Cancel Signals



More UML Activity Diagram Notation

How to show conditional branches?

More UML Activity Diagram Notation

- How to show conditional branches?
 - The decision symbol and the related merge symbol shows how flows can come back together.

Decision and Merge



NextGen POS (Process Sale) Use Case



Quiz

- Why are State Diagrams Useful?
- What is transition action, guard condition and nested state used for in State Diagrams?
 Provide examples.
- What is partitioning used for in Activity Diagrams?
- What is the meaning of rake symbol and signals in Activity Diagrams?

Actions

- Review Slides
- Read Chapter 29, Modeling Behavior in Statechart Diagrams
 - Applying UML and Patterns, Craig Larman