

Events

Object-Oriented Analysis and Design
CSCI 6448 - Fall 1998
Kenneth M. Anderson

Goals of this Lecture

- Present the notion of an event
 - Definition
 - Relationship to object-state
 - Relationship to operations

Events

- Last time
 - An object's state is a collection of attributes and relationships for a particular object
 - A state change is the time when one of these relationships or attributes changes
- An event is a notification of a state change

Events, continued

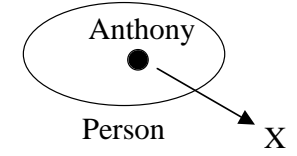
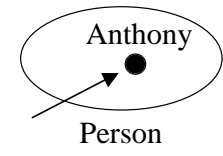
- In particular
 - Events are notifications of **IMPORTANT** state changes
 - Important means
 - We wish to record the state change
 - We wish to react to the state change
 - We wish to gain information from the state change

Types of Events

- Fundamental Events
 - Add
 - Remove
- Basic Events
 - Creation
 - Termination
 - Classification
 - Declassification
 - Disconnection
- Compound Events
 - Reclassification
 - Reconnection
 - Component termination
 - Coalesce
 - Decoalesce
- Compound events require both adds and removes with transactional semantics

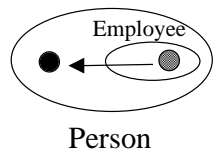
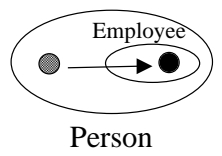
Creation and Termination

- Creation
 - Occurs when an object is created and becomes a member of a set
- Declassification
 - Occurs when an object is deleted or becomes a member of no sets



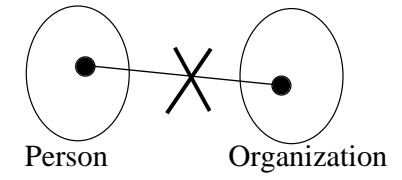
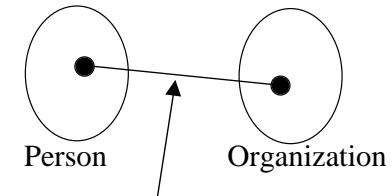
Classification + Declassification

- Classification
 - Occurs when an object's membership in class sets is increased by one
- Declassification
 - Occurs when an object's membership in class sets is decreased by one



Connection and Disconnection

- Connection
 - Adds a new relationship between two objects
- Disconnection
 - Removes an existing relationship between two objects





Reclassification

- The simultaneous declassification of an object as one type and classification of it as another type
- A marriage event reclassifies someone from being unmarried to being married in one event
 - Transactional semantics means that at all times the object is one or the other



Reconnection

- A reconnection is the simultaneously termination of one tuple and the creation of another tuple of the same relationship
- These events help preserve cardinality constraints with a minimum cardinality of one (i.e. mandatory)



Other events

- Component Termination
 - Refers to the deletion of a composite object; the object and all of its parts are simultaneously deleted
- Coalesce/Decoalesce
 - Objects that once were thought distinct are shown to be the same object; or vice-versa
 - “The Butler did it!”



Event pre- and post-conditions

- An event precondition is a state that must apply to an object before the event occurs
- An event post-condition is a state that must apply to an object after the event occurs
- That is why events are associated with the transitions of state diagrams



Event causation

- An event does not happen spontaneously; they have a source or cause that generates them
- Three types of event generators
 - Internal
 - External
 - Temporal



Definitions

- An internal event is the result of an operation or activity within the domain of the project
- An external event is the result of an activity outside of the project domain
- A temporal event are the result of clock operations
- Clock operations emit a specified pattern of clock-tick events
- They can be both internal or external



Event Occurrences and Events

- An event occurrence is a noteworthy change in state
- An event is a type of event occurrence
- Event: Airplane Departs
- Event Occurrence: Delta Flight 853 departs DIA for SNA on June 3, 1998 at 12:53 PM MDT.



Object History

- Every event affects an object and is the result of an operation
- Thus, an object's history
 - Is a record of all of its event occurrences
- This is another way of defining the term “complete state” from last lecture.



Event Partitions

- The section on event partitions in Chapter 13 of Martin and Odell describe one of the precursors to activity diagrams: event diagrams
- Note: the use of decision points and guarded transitions



How to use events

- Events are one of the first steps towards modeling behavior in your system
- Based on your class diagram
 - Identify important state changes
 - What changes should trigger other activities in your system?
 - These triggers are your project's events



Using Events, continued

- Use these events to drive activity diagrams and state diagrams
 - Events label the transitions of these diagrams
- As you iterate, additional events will allow you to supply additional detail to these diagrams