**Chapter 6 Behavioral Modeling  
Testbank**

**Multiple Choices**

1. The external behavior of a system is described by \_\_\_\_\_.

a. functional models

b. structural models

c. behavioral models

d. interaction diagrams

e. statechart diagrams

Ans: a

1. An analyst depicts the static view of an information system with \_\_\_\_\_.

a. use-case models

b. structural models

c. behavioral models

d. interaction diagrams

e. statechart diagrams

Ans: b

1. The two types of interaction diagrams are \_\_\_\_\_\_\_\_\_\_\_\_\_\_ diagrams.

a. use-case and sequence

b. class and sequence

c. sequence and communication

d. object and communication

e. statechart and object

Ans: c

1. The modeling focus of the interaction diagram is at the \_\_\_\_\_ level while the modeling focus on the class diagram is at the \_\_\_\_\_ level.

a. class, object

b. interaction, class

c. class, interaction

d. object, class

e. high, low

Ans: d

1. A(n) \_\_\_\_\_ is an instantiation of a class.

a. attribute

b. behavior

c. operation

d. message

e. object

Ans: e

1. A(n) \_\_\_\_\_ describes information about an object.

a. attribute

b. behavior

c. operation

d. message

e. instance

Ans: a

1. \_\_\_\_\_ are information that is sent to objects to tell it to execute one of its behaviors.

a. Attributes

b. Operations

c. Messages

d. Instances

e. use-cases

Ans: c

1. Analysts use \_\_\_\_\_ to show the objects that participate in a use case and the messages that pass between the objects over time for one particular use case.

a. structural models

b. sequence diagrams

c. collaboration diagram

d. class diagrams

e. behavioral state machines

Ans: b

1. When arranging actors and objects on a sequence diagram, it is nice to list them \_\_\_\_\_.

a. in alphabetical order down the side of the diagram

b. in alphabetical order across the top of the diagram

c. in order in which they participate in the sequence down the side of the diagram

d. in order in which they participate in the sequence across the top of the diagram

e. actors and objects are not both shown on a sequence diagram

Ans: d

1. Which of the following objects would be most likely to be destroyed at some point in time in a sequence diagram?

a. customer

b. order

c. order item

d. invoice

e. shopping cart

Ans: e

1. A systems analyst draws a lifeline with an X at the end. This lifeline represents \_\_\_\_\_.

a. a message that cannot be delivered

b. an object that is destroyed at a point in time

c. a message that is delivered at that point in time

d. an object that is complete at that point in time

e. an object that arrives at its final destination

Ans: b

1. The focus in a sequence diagram is on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

a. How actors interact with objects to realize a given use case

b. Messages sent by actors to other objects

c. when an object is being created

d. when messages are being destroyed

e. time ordering of messages being passed between objects

Ans: e

1. On a sequence diagram, an object name of Students:List would indicate that \_\_\_\_\_.

a. Students is an instance of the List class that contains individual student objects

b. List is an instance of the Students class

c. List is a method of the Students class

d. the Students and Lists objects are combined for that step in the sequence diagram

e. a message is being passed from the Students class to the List class

Ans: a

1. The order of messages on a sequence diagram goes from \_\_\_\_ to \_\_\_\_\_.

a. right to left

b. bottom to top

c. left to right

d. top to bottom

e. left to right, top to bottom

Ans: d

1. When an object sends a message to itself in a sequence diagram that is referred to as \_\_\_\_\_.

a. recursive-messaging

b. self-messaging

c. self-delegation

d. recursive-delegation

e. none of these

Ans: c

1. The first step in building a sequence diagram is to \_\_\_\_\_.

a. set the context

b. identify which objects will participate

c. set the lifeline for each object

d. add the focus of control to each object’s lifeline

e. validate the sequence diagram

Ans: a

1. The acronym CRUD stands for \_\_\_\_\_.

a. create, read, update, delete

b. create, reference, underline, delete

c. create, re-do, underline, delete

d. collaborate, read, update, delete

e. collaborate, reference, update, discard

Ans: a

1. A behavioral state machine is a dynamic model that shows the different state through which a single \_\_\_\_\_ passes through its life in response to events, along with its responses and actions.

a. object

b. actor

c. use case

d. sequence diagram

e. communication diagram

Ans: a

1. A set of rules are used for performing the verification and validation walkthrough of behavioral model. One of the rules is: if there is a message on the sequence diagram, there must be \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_on the communications diagram, and vice versa.

a. a communication

b. a message

c. a behavior

d. a dependence

e. an association

Ans: e

1. A set of rules are used for performing the verification and validation walkthrough of behavioral model. One of the rules is: all entries in a CRUDE matrix imply \_\_\_\_\_\_\_\_\_\_\_\_\_ being sent from an actor or object to another actor or object.

a. a message

b. a transition

c. an association

d. a guard condition

e. a behavior

Ans: a

1. A set of rules are used for performing the verification and validation walkthrough of behavioral model. One of the rules is: all \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ contained in a behavior state machine must be associated with a message being sent on a sequence and communication diagram, and it must be classified as a (C)reate, (U)pdate, or (D)elete message in a CRUDE matrix.

a. messages

b. behaviors

c. transitions

d. associations

e. dependences

Ans: c

**True/False**

1. Modeling the real world in behavioral models is relatively easy.

Ans: False

1. As behavioral models are developed, it is likely that the use cases and structural models will have to be modified.

Ans: True

1. Each object can both send and receive messages.

Ans: True

1. A sequence diagram can only be drawn to represent a single scenario in a use case.

Ans: False

1. Actors and objects are placed on a sequence diagram in no particular order.

Ans: True

1. The destruction of temporary objects is shown on sequence diagrams with an X at the end of the lifeline.

Ans: True

1. When an object sends a message to itself, it is called self-messaging.

Ans: False

1. It is possible for an object to create another object, resulting in the message going directly to the object, not its lifeline.

Ans: True

1. If you are interested in the time ordering of the messages in a system, you should use a communication diagram.

Ans: False

1. Both the communication diagram and the sequence diagram can show return messages.

Ans: False

1. A message with the information [a Student exists] will be sent only when this condition is true.

Ans: True

1. The first step in drawing a communication diagram is to identify which objects and associations between the objects participate in the communication.

Ans: False

1. In a CRUD matrix, C represents Create.

Ans: True

1. When creating a CRUD matrix, you should use the letter U to represent Underline.

Ans: False

1. CRUD matrices are created by creating a matrix that lists the classes across the top and down the side.

Ans: True

1. Conditions on conditional messages in communication diagrams are placed in [ ] symbols.

Ans: True

1. Adding messages is the last step in building a communication diagram?

Ans: False

1. If a class represents only temporary objects, then there should be a D in the column of the CRUD matrix.

Ans: True

1. CRUD matrices are useful in the step “Set the Context” of building a collaboration diagram?

Ans: False

1. Events are the triggers that cause an object to move from one state to another.

Ans: True

1. A guard condition is a Boolean expression, which allows a transition to occur only if a condition is satisfied.

Ans: True

1. The following symbol represents the destruction of a temporary object in a sequence diagram.

Ans: True

1. An execution occurrence in a sequence diagram denotes when an object is sending or receiving messages.

Ans: True

1. An execution occurrence in a sequence diagram is represented using thick dashed lines with arrows that are placed on top of the life lines.

Ans: False

1. Return messages in a sequence diagram are often shown using dashed lines with arrows labeled with the return value.

Ans: True

1. One of the guidelines for creating sequence diagram is: Place the initiator of the scenario on the right of diagram.

Ans: False

1. The communication diagram is used for modeling process flow.

Ans: False

1. Only create behavioral state machine for “complex” objects.

Ans: True

1. Using detail and descriptive names for states when creating behavioral state machine.

Ans: False

1. Every actor and object included on a sequence diagram must be included as an actor and an object on a communication diagram, and vice versa.

Ans: True

1. If there is a message on the sequence diagram, there must be an association on the communications diagram, and vice versa.

Ans: True

1. Some entries in a CRUDE matrix does not imply a message being sent from an actor or object to another actor or object.

Ans: False

1. System analysts use behavioral models to depict the internal structural or static view of an information system.

Ans: False

1. System analysts use structural models to depict the internal structural or static view of an information system.

Ans: True

**Short Answer**

1. Describe sequence diagrams.

Ans: Sequence diagrams are one of the types of interaction diagrams (the other being communication diagrams). These diagrams show the objects that participate in a use case and also the time-based ordering of the messages that pass between those objects over time for a single use case. A sequence diagram is a dynamic model that shows explicit sequence of messages between objects in a defined interaction. A sequence diagram can be a generic diagrams which shows all possible scenarios for a given use case or an instance sequence diagram which depicts a single scenario within the use case. Different components of a sequence diagram include actors, objects, lifelines, messages, execution occurrences, and frames.

1. Describe communication diagrams.

Ans: Communication diagrams are also a type of interaction diagrams. These diagrams are essentially object diagrams that show message passing relationships instead of associations. These diagrams help you see the pattern of activity that occurs over a set of collaborating classes. Unlike sequence diagrams, they are unable to indicate time-ordering of messages. Primary components of a communication diagram include actors, objects, and messages.

1. What is a sequence diagram used for? Why would an analyst choose a sequence diagram over a communication diagram?

Ans: A sequence diagram is an interaction diagram (as is the communication diagram) that looks at the time-based order of the messages among the objects in the use case. Sequence diagrams are constructed by placing the objects across the top of the diagram, and drawing the messages from lifeline to lifeline, from top to bottom to reflect the time-based order of the messages. Communication diagrams are essentially object diagrams that show message passing relationships instead of associations. These diagrams help you see the pattern of activity that occurs over a set of collaborating classes. Unlike sequence diagrams, they are unable to indicate time-ordering of messages. An analyst would choose to draw a sequence diagram if this time-based order is important to the analysis of the use case under study.

1. When drawing a sequence diagram, the analyst uses lifelines and a symbol to represent the execution occurrence. It is also possible for an object to be destroyed at some point in time. Explain these three concepts – what is a lifeline, what does the execution occurrence mean, and how it is possible that an object can be destroyed in a use case. Show the diagramming symbols used to represent each of these concepts.

Ans: A lifeline extends vertically down from the object, representing literally its life in the system. Most objects have an unlimited life in the use case, so there is nothing that denotes an end to its life (obviously it ends at the end of the diagram). The lifeline is drawn as a dashed vertical line.

An execution occurrence symbol is placed on the lifeline to represent when that particular object is sending or receiving messages. Thus, it shows when, in the life of the object, it is the focus of the use case. The focus of control is drawn as a long, narrow rectangle that is placed on top of the vertical lifeline.

An object may be destroyed in a use case if it is a temporary object. For example, a web-based shopping cart will disappear once checkout occurs. If there is a temporary object, the lifeline is shortened, and an X is placed at the bottom of the lifeline to represent the end of the life.

1. List the steps that an analyst should go through to build a sequence diagram.

Ans:

1. Set the context – is it a system, a use case, a scenario of a use case, or an operation of a class
2. Identify which objects will participate
3. Set the lifeline for each object
4. Lay out the messages from the top to the bottom of the diagram based on the order in which they are sent
5. Add the execution occurrence to each object’s lifeline
6. Validate the sequence diagram
7. How does a communication diagram differ from a sequence diagram? Why would an analyst choose to draw a communication diagram?

Ans: A communication diagram emphasizes the flow of messages through a set of objects, while the sequence diagram focuses on the time ordering of the messages that are being passed. If the focus of the analyst is on understanding how a set of collaborating objects interact with one another, then a communication diagram is more appropriate. If the time sequence of the message were of interest to the analyst, then he or she would create a sequence diagram.

1. Identify the steps that an analyst would follow in building a communication diagram.

Ans:

1. Set the context, which may be a system, a use case, a scenario of a use case, or an operation of a class
2. Identify which objects (actors) and the associations between the objects that participate in the collaboration
3. Lay out the communication diagram
4. Add the messages
5. Validate the communication diagram
6. What does the acronym CRUD stand for? Why would an analyst create a CRUD matrix during his analysis?

Ans: CRUD stands for Create, Read or Reference, Update, and Delete. These are the four possible interactions between classes in a system. An analyst would create a CRUD matrix to help identify potential collaborations while building a communication diagram. (Step 2 of building a communication diagram). CRUD matrix clearly identifies how objects create, reference, modify and delete other objects.

1. When looking at a CRUD matrix, why would it be important to see “D” in a column for a temporary object, or to not see “D” or “U” in a column for objects in a data warehouse?

Ans: D represents delete, and U represents update. A temporary object is supposed to disappear at some point in the life of the system; if there were no D in the CRUD matrix, then the analyst failed to accurately model that particular object.

A data warehouse is supposed to maintain historical information, so one would not expect to see the data deleted (D), and in most cases, it should not be updated either (U), since the data warehouse should be reflecting the historical values, not updated ones. Hence, objects that represent a data warehouse should not have U or D underneath them in the CRUD matrix.

1. Think of an employee object. What are several of the possible states that that object may have over time? Which of these states would be the initial state? Which would be the final state?

Ans:

* New Employee (initial state)
* Former Employee (final state)
* Current Employee
* Probationary Employee

1. Explain the term transition.

Ans: A transition is a relationship that represents the movement of an object from one state to another. Some state transitions are conditional. Such conditional transitions are associated with a guard condition. A guard condition is a Boolean expression that includes attribute values, and it allows a state transition to occur only if the condition is true. An object moves from one state to another based on the outcome of an action that is triggered by an event.

1. Explain the terms event, action and activity.

Ans: An event is something that takes place at a certain point in time and changes value(s) that describe an object. This in turn causes a change in the state of the object. Examples of events include a condition becoming true, a receipt of a call fro a method by an object, and the passage of a designated period of time. An action is an atomic, non-decomposable process that cannot be interrupted. Actions are often modeled as if they are completed in zero time, and they are associated with a transition. An activity is a non-atomic, decomposable process that can be interrupted. Activities take a long period of time to complete, they can be started or stopped by actions, and they are associated with states.

17. How are guard conditions shown on a behavioral state machine?

Ans: A guard condition is a Boolean expression that includes attribute values, which allows a transition to occur only if the condition is true.

The Boolean expression is placed in brackets and located along the relevant transition.

1. What kinds of events can lead to state transitions on a behavioral state machine?

Ans: In the parlance of behavioral state machines, every event leads to a state transition. This is because events are defined as anything that changes a value which in turn describes the state of the behavioral state machine. In other words, transitions occur only as the result of an event. Specific types of events are as follows.

1. Data value change: a change in one of the data values that collectively describe the state of the object. For example, a patient transitions from “new” to “existing” after the first visit.
2. Boolean condition: a Boolean test that is applied at a certain point. For instance, after a patient transitions to “existing” we may ask the question “is the patient insured?” which would lead to state changes based on billing policies.
3. Time lapse: after a certain amount of time in a given state, the object may spontaneously change to a different state. For instance, a patient that is in state “inactive” (perhaps because he/she has no future appointments), may be removed from the system after 5 years.