**Chapter 6 Structural Modeling  
Testbank**

**Multiple Choices**

1. Which of the following are used to create objects?

a. concrete objects

b. abstract objects

c. concrete classes

d. abstract classes

e. concrete instances

Ans: c

1. A(n) \_\_\_\_\_ of an analysis class represents a piece of information that is relevant to the description of the class.

a. instance

b. object

c. attribute

d. operation

e. relationship

Ans: c

1. Which of the following would most likely **not** be an example of an attribute?

a. employee name

b. customer address

c. stock number

d. ISBN number

e. cancel appointment

Ans: e

1. Which of the following types of attributes is **not** proper in an analysis class?

a. integers

b. strings

c. doubles

d. compound

e. date

Ans: d

1. A(n) \_\_\_\_\_ of an analysis class is where the behavior of the class is defined.

a. operation

b. attribute

c. class

d. object

e. abstract class

Ans: a

1. Service is another name for \_\_\_\_\_.

a. attribute

b. class

c. abstract class

d. operation

e. object

Ans: d

1. “A-kind-of” relationships represent \_\_\_\_\_ relationships.

a. generalization

b. association

c. aggregation

d. subsetting

e. vague

Ans: a

1. “A-part-of” or “has-parts” relationships represent \_\_\_\_\_ relationships.

a. generalization

b. association

c. aggregation

d. subsetting

e. vague

Ans: c

1. If a “student signs up for a class,” which type of relationship would you use to model the relationship between the two?

a. generalization

b. association

c. aggregation

d. subsetting

e. vague

Ans: b

1. CRC cards are used to document the responsibilities and collaborations of a(n) \_\_\_\_\_.

a. class

b. relationship

c. object

d. attribute

e. operation

Ans: a

1. A(n) \_\_\_\_\_ formalizes the interactions between a client and server object.

a. relationship

b. contract

c. abstract object

d. concrete object

e. abstraction

Ans: b

1. A collaboration consists of \_\_\_\_\_.

a. two instances of a class talking with each other

b. two instances of a class knowing the value of each other’s attributes

c. a set of classes that share common operations

d. a set of classes that are all related to one another

e. a set of classes involved in a use case

Ans: e

1. A \_\_\_\_\_ object is an instance of a class that sends a request to an instance of another class for an operation to be executed.

a. client

b. server

c. contract

d. requester

e. CRC

Ans: a

1. A \_\_\_\_\_ object is the instance of a class that receives a request from another object.

a. client

b. server

c. contract

d. provider

e. CRC

Ans: b

1. A mechanism for developing CRC cards is for the user or analyst to role-play as if they are an instance of the class. This process is called \_\_\_\_\_.

a. acting

b. anthropomorphism

c. interviewing

d. anamorphous

e. observation

Ans: b

1. Which of the following is part of a CRC card?

a. class name

b. type

c. description

d. responsibilities

e. all of these

Ans: e

1. Which of the following would **not** be an appropriate class name?

a. student

b. patient

c. John

d. customer

e. doctor

Ans: c

1. A class diagram is a(n) \_\_\_\_\_ model.

a. static

b. dynamic

c. evolving

d. obsolete

e. none of these

Ans: a

1. A public attribute is shown in a class diagram with the symbol \_\_\_\_\_ before the name.

a. –

b. #

c. \*

d. +

e. /

Ans: d

1. One way to identify objects for the class diagram is to scan the use case descriptions for \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

a. relationships

b. object lists

c. patterns

d. nouns

e. verbs

Ans: d

1. Many different approaches have been suggested to aid the analyst in identifying a set of candidate objects for the structural model. The four most common approaches are textual analysis, brainstorming, common object lists, and \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

a. use cases

b. business models

c. patterns

d. nouns

e. verbs

Ans: c

1. Text analysis is one of the four approaches to aid the analysis in identifying a set of candidate objects for the structural model. Textual analysis is an analysis of the text in the \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

a. business models

b. use case descriptions

c. class diagrams

d. activity diagrams

e. use case diagrams

Ans: b

1. This chapter suggested three representations that could be used for structural modeling: CRC cards, class diagrams, and \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

a. object diagrams

b. use case diagrams

c. activity diagrams

d. use case description

e. business models

Ans: a

1. One of rules for verifying the structural model is to check that the object type of the attributes listed on the back of the CRC card and with the attributes in the attribute list of the class on a class diagram implies \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_from the class to the class of the object type.

a. an inheritance

b. an aggregation

c. a dependence

d. an association

e. a communication

Ans: d

**True/False**

1. A structural model is a formal way of representing the business activities in a business system.

Ans: False

1. Abstract classes are used to create objects.

Ans: False

1. Aggregation relationships can be useful for both aggregation and decomposition.

Ans: True

1. Association relationships are used for the relationship between instances of two classes when the relationship is not specific enough to be considered a generalization or an aggregation relationship.

Ans: True

1. CRC cards are used to capture all of the relevant information associated with a class.

Ans: True

1. The responsibilities of a class can be divided into knowing and doing types.

Ans: True

1. A contract formalizes the interactions between one client object and other client objects.

Ans: False

1. Role-playing an instance of a class is a good way to develop CRC cards.

Ans: True

1. A class diagram is a dynamic model that shows how the classes and relationships change over time.

Ans: False

1. The symbol for a class in a class diagram contains the name of the class, a list of the attributes, and a list of operations.

Ans: True

1. It is possible for the multiplicity of a relationship to be “zero or more.”

Ans: True

1. Views are often used to provide the user a complete picture of the class diagram.

Ans: False

1. Object diagram is just another name for class diagram; they both show the same information.

Ans: False

1. A common object list is a list of objects that are common to business domains.

Ans: True

1. Patterns can be reused in the development of object-oriented systems.

Ans: True

1. Textual analysis is the review of corporate documents to identify potential objects, attributes, operations, and relationships.

Ans: False

1. Patterns are useful for identifying classes, attributes, operations, and relationships in certain business domains.

Ans: True

1. Part of the process of creating CRC cards is role-playing them, where individuals perform the operations for the class on the card.

Ans: True

1. Books have been written that list useful patterns for a wide variety of business domains.

Ans: True

1. The main building blocks of a class diagram are the relationships between entities

Ans: False

1. In the symbol for a class, the name goes in the bottom compartment.

Ans: False

1. A class symbol has the attributes listed in the top compartment.

Ans: False

1. Operations of a class are listed in the bottom compartment of the class symbol.

Ans: True

1. Age is an example of a derived attribute in a class since it can be computed from the date-of-birth attribute and the current date.

Ans: True

1. A derived attribute that is stored is shown on a class diagram with the symbol # before the name.

Ans: False

1. The symbol - before the name of an attribute on a class diagram represents a private attribute.

Ans: True

1. The default visibility for an attribute in a class diagram is public.

Ans: False

1. The default visibility of an operation in a class diagram is public.

Ans: True

1. A constructor operation creates a new instance of a class?

Ans: True

1. The update operation makes information about the state of an object available to other objects without altering the object.

Ans: False

1. The query operation will change the value of one or more of an object’s attributes.

Ans: False

1. The multiplicity one-or-more on a relationship is signified by the notation 1..M.

Ans: False

1. The multiplicity zero or more is signified by the notation 0..\*.

Ans: True

1. \*..1 is a valid multiplicity on a relationship in a class diagram.

Ans: False

1. When classes share a many-to-many relationship, it is common to create an association class that has its own attributes and methods.

Ans: True

1. A diamond at the end of a relationship in a class diagram represents the many side of the relationship.

Ans: False

1. A view is used to simplify a class diagram by subsetting the information available on the diagram for viewing.

Ans: True

1. An object diagram may be created to make a class diagram easier to read by grouping classes together.

Ans: False

1. In the textual analysis of a use case, the use case name suggests possible classes while the verbs suggest possible operations.

Ans: False

1. In textual analysis of a use case, a doing verb implies an operation.

Ans: True

1. A pattern is a group of collaborating classes that solve a recurring problem.

Ans: True

1. Common object lists technique identifies candidate objects for a structural model by analyzing the text in the use-case descriptions.

Ans: False

1. Common object lists technique identifies candidate objects for a structural model by analyzing a list of objects that are common to the business domain.

Ans: True

1. The four most common approaches have been suggested to aid the analyst in identifying a set of candidate objects for the structural model are textual analysis, brainstorming, common object lists, and design patterns.

Ans: False

1. Textual analysis to aid the analyst in identifying a set of candidate objects for the structural model is an analysis of the text in the business model.

Ans: False

1. CRC cards, class diagrams, and object diagrams are three representations could be used for structural modeling.

Ans: True

1. The set of rules used for verifying and validating the structural model is to check the consistency between CRC cards and class diagrams of the structural model.

Ans: True

1. One of the rules used for verifying and validating the structural model is to check that every CRC card should be associated with a class on the class diagram, and vice-versa.

Ans: True

1. One of the rules used for verifying and validating the structural model is to check that the object type of the attributes listed on the back of the CRC card and with the attributes in the attribute list of the class on a class diagram implies an aggregation from the class to the class of the object type.

Ans: False

1. One of the rules used for verifying and validating the structural model is to check that the responsibilities listed on the front of the CRC card must be included as attributes in a class on a class diagram, and vice versa.

Ans: False

**Short Answer**

1. Define what a structural model is. Why should a systems analyst create one? Why is the creation of a structural model iterative?

Ans: A structural model is a formal way of representing the data that are used and created by a business system. It shows the people, places, or things about which information is captured and used by the system, and it shows the interrelations between them.

Without creating a structural model, the analyst will likely not have a complete understanding of the data that are key to the application domain. In addition, the structural model helps assure that the analyst is talking the same language as the users.

Development of a structural model is iterative because it evolves as the analyst gains understanding of the application domain. The structural model is high-level (conceptual) during the analysis phase, and becomes more concrete and detailed as the project moves through the SDLC.

1. Define the following terms: class, attribute, and operation. Give examples of each.

Ans: A class is a template used to create specific instances or objects in the application domain. All of the objects of a given class will have the same structure and behavior, but will vary in the data in their attributes. For example, a class may be “student” and the instances of that class would be specific students – John, Jane, Sue – where the attribute first name obviously has a different value for each student.

An attribute represents a piece of information that is relevant to the description of the class. They contain information the system should store for each class. For the student class mentioned above, possible attributes would include first name, last name, address, student ID number, gender, classification, etc.

An operation is where the behavior to the class is defined. These are the actions to which the instances of the class can respond. Examples would include “drop student from class” and “alter student name.”

1. An analyst has discovered two classes where one appears to be a superclass and the other a subclass. What type of relationship is used to model this association on a structural model? What is the notation that is used? Draw an example of how this would be illustrated on a class diagram.

Ans: When there is a superclass and one or more subclasses, a generalization relationship is used to represent the “a-kind-of” relationship. The notation is a solid line from the subclass to the superclass and a hollow arrow pointing at the superclass. An example is shown below. Employee superclass has Data Entry Clerk as the subclass.

1. What is a CRC card and why should an analyst create them? What are the essential elements of a CRC card?

A CRC card is a Class-Responsibility-Collaboration card that is used to document the responsibilities and collaborations of a class. They should have all the information needed for the analyst to draw the logical structural model for the system. Each card captures and describes the essential elements of a class. The front of the card as shown in the text contains: the class name, ID, type, description, list of associated use cases, responsibilities, and collaborators. The back of the CRC card contains: the attributes and relationships of the class.

1. What are the three parts of the notation for a class on a class diagram? Define each part.

Ans: The class notation on the class diagram contains a name, which should be a noun, in the top compartment, the list of attributes in the middle compartment, and a list of operations in the bottom compartment. The attributes are characteristics of the class that will vary in value for each instance of the class (e.g., first name could be an attribute but it would have a different value for each instance of the class). The operations are actions or functions that the class can perform, such as “prepare back-order notice” or “drop student from class.”

1. What is a relationship? What are the types of relationships that you may have on a class diagram? Define each of these.

Ans: A relationship is the association between two classes on a class diagram. There are at least three types of relationships that occur on class diagrams: normal, generalization, and aggregation. A normal relationship is one that does not involve either generalization or aggregation. Thus, it is drawn between classes that are not subclass/superclass (generalization) or where one of the classes is a subset of another (aggregation).

A generalization relationship is drawn between a superclass and a subclass, showing the inheritance of properties and operations from the superclass by the subclass. An example would be Employee as the superclass and Data Entry Clerk as the subclass. This type of relationship is described by the words “is a-kind-of.”

An aggregation relationship is used when a group of classes actually comprise another class. An example would be the class Students is actually comprised of the classes freshmen, sophomores, juniors, and seniors. These relationships are often described as “is a part of” or “is made up of.”

1. Describe how views and packages help simplify class diagrams.

Ans: Views are a subset of the class diagram that only shows the information desired by the user. For example, it may show only the classes and relationships that are relevant to a particular case, or only show the relationships of a particular type.

Packages are used to group classes together that are related logically. This reduces the level of complexity and simplifies the diagram by reducing the number of elements on the diagram.

1. Describe the textual analysis method for identifying candidate objects for a structural model.

Ans: Textual analysis involves analyzing the text of the use cases, using the text to provide clues about classes, instances of classes, attributes, operations, and relationships. The analyst starts by reviewing the use-case descriptions and use-case diagrams. The text in the descriptions is examined to identify potential objects, attributes, operations and relationships. Nouns in the use cases suggest possible classes, while verbs suggest possible operations. The simplicity of this technique is a major advantage.

1. Describe the common object list technique for identifying candidate objects for a structural model.

Ans: Common object lists are simply a list of objects that are common to the business domain. This involves the analyst ignoring the use cases and looking instead for physical or tangible things in the business domain, incidents that occur, or roles that people play in the business domain. Each of these can help identify objects that were missed in the analysis of the use cases.

1. Describe the patterns method for identifying candidate objects for a structural model.

Ans: Patterns is the third way to identify candidate objects for a structural model. This involves matching exiting patterns of classes to the target business domain to see if any of them can be reused. Books of patterns have been published for many of the common business domains, and they can be referenced by the analyst as a way of identifying possible objects. By simply reusing the existing patterns of classes, an analyst can define the system more quickly and more completely. Many common patterns have been published ranging from high-level business oriented patterns to low-level design patterns.

1. Describe how the concept of patterns in architecture relate to the development of object-oriented systems. In what way do patterns promote reuse?

Christopher Alexander, an architect, contends that very sophisticated buildings can be constructed by stringing together commonly found patterns in existing buildings, rather than creating entirely new concepts and designs every time an architect desires to design a building. In the same way, patterns can be used in the development of object-oriented systems because many problems in the business domain involve the same type of objects and interactions as similar problems in other systems. Once the pattern has been developed, it can be reused in many other similar systems. One example the text provided was the example of a business transaction. Virtually all transactions would require a transaction class, a transaction line item class, an item class, a location class, and a participant class. Thus, once the analyst has determined that a transaction is involved, he has identified all of those classes that need to be considered in the development of this system.

1. List and briefly describe the seven steps for object identification and structural modeling.

The seven steps are:

1. Create CRC cards by performing textual analysis on the use cases
   * Create the normal flow of events, subflows, and alternate and exceptional flows following the SVDPI format
2. Brainstorm additional candidate classes, attributes, operations, and relationships by using the common object list approach
   * This may result in revisions to the CRC cards as additional elements are discovered
3. Role-play each use case using the CRC cards
   * Assign an individual to perform the operations for the class; if the “system” breaks down, you will likely discover additional objects, attributes, operations, or relationships
4. Create the class diagram based on CRC cards
   * Transfer the information from the CRC cards to the class diagram, looking for ways in which to improve the model with generalization or aggregation relationships; keep the defaults of private attributes and public operations unless there is a reason to do differently
5. Review the structural model for missing and/or unnecessary classes, attributes, operations, and relationships
6. Incorporate useful patterns
   * This step requires the analyst to identify the underlying problem domain and to search for matching patterns that can be used to compare to what is already in the model
7. Review the structural model
   * Do this step as part of a formal walk-through with a team of developers and users
8. Explain the concept of object diagrams. How do they differ from class diagrams?

Ans: An object diagram is essentially an instantiation of all or part of the class diagram. Object diagrams are very useful when one is trying to uncover the details of a class. This is because it is easier to think in terms of concrete objects (instances) than in an abstract way using classes. By looking at the concrete objects (instances), an analyst may uncover more classes or methods or attributes that should belong to classes.

1. Describe different types of multiplicity that may be found on relationships in a class diagram.

Ans: (1) Exactly one. An example of this type of multiplicity is that each Department has one and only one Manager. (2) Zero or More. An example is that an Employee may have zero or more Dependents. (3) One or More. An example is that a Manager is responsible for one or more Employees (4) Zero or One. An example is that an Employee has zero or one Spouse. (5) Specified Range. An example is that an employee can take between two to four vacations each year. (6) Multiple disjoint ranges. An employee may be a member of one to three or five committees.

1. Describe the four most common approaches have been suggested to aid the analyst in identifying a set of candidate objects for the structural model. These methods are textual analysis, brainstorming, common object lists, and patterns.

Ans: Textual analysis is an analysis of the text in the use-case descriptions. The analyst starts by reviewing the use-case descriptions and the use-case diagrams. The text in the descriptions is examined to identify potential objects, attributes, operations, and relationships. The nouns in the use case suggest possible classes, whereas the verbs suggest possible operations.

Brainstorming is a discovery technique that has been used successfully in identifying candidate classes. Essentially, in this context, brainstorming is a process of a set of individuals setting around a table and suggesting potential classes that could be useful for the problem under consideration. Typically, a brainstorming session is kicked off by a facilitator that asks the set of individuals to address a specific question or statement that “frames” the session.

A common object list is simply a list of objects common to the business domain of the system. Several categories of objects have been found to help the analyst in the creation of the list, such as physical or tangible things, incidents, roles, and interactions.4 Analysts should first look for physical, or tangible, things in the business domain. These could include books, desks, chairs, and office equipment. Normally, these types of objects are the easiest to identify. Incidents are events that occur in the business domain, such as meetings, flights, performances, or accidents. Reviewing the use cases can readily identify the roles that the people play in the problem, such as doctor, nurse, patient, or receptionist. Typically, an interaction is a transaction that takes place in the business domain, such as a sales transaction. Other types of objects that can be identified include places, containers, organizations, business records, catalogs, and policies. In rare cases, processes themselves may need information stored about them. In these cases processes may need an object, in addition to a use case, to represent them. Finally, there are libraries of reusable objects that have been created for different business domains.

A pattern is simply a useful group of collaborating classes that provide a solution to a commonly occurring problem. Because patterns provide a solution to commonly occurring problems, they are reusable.

1. Describe the set of rules for verifying and validating the structural model.

Ans: Three representations could be used for structural modeling: CRC cards, class diagrams, and object diagrams. Because an object diagram is simply an instantiation of some part of a class diagram, we limit our discussion to CRC cards and class diagrams. We need a set of rules that will test the consistency within the structural models.

First, every CRC card should be associated with a class on the class diagram, and vice-versa.

Second, the responsibilities listed on the front of the CRC card must be included as operations in a class on a class diagram, and vice versa.

Third, collaborators on the front of the CRC card imply some type of relationship on the back of the CRC card and some type of association that is connected to the associated class on the class diagram.

Fourth, attributes listed on the back of the CRC card must be included as attributes in a class on a class diagram, and vice versa.

Fifth, the object type of the attributes listed on the back of the CRC card and with the attributes in the attribute list of the class on a class diagram implies an association from the class to the class of the object type.

Sixth, the relationships included on the back of the CRC card must be portrayed using the appropriate notation on the class diagram.

Seventh, an association class should be created only if there is indeed some unique characteristic (attribute, operation, or relationship) about the intersection of the connecting classes.

Finally, some specific representation rules must be enforced.

1. For what purpose is an association class used in a class diagram? Give an example of an association class that may be found in a class diagram that captures students and the courses they have taken.

Ans: The association class is used to show a relationship where the relationship itself has attributes -- in other words which has information regarding the relationship itself that should be captured. Students enroll in courses. There is information about that enrollment itself -- when it occurs and the grade received, for example, that would be of interest to keep track of.