CHAPTER 5: STRUCTURAL MODELING

In the previous chapter's installment of the CD Selections case, we saw how Alec, Margaret, and the team worked through building functional models of the business processes (see Figures 4-A through 4-G) contained in their evolving Web-based solution. In this chapter, we introduced how structural models using CRC cards and class and object diagrams could be created, verified, and validated. In this installment of the CD Selections case, we see how Alec and Margaret work through creating, verifying, and validating the structural models of the Web-based solution based on the one completed detailed, essential use case: Place Order (see Figure 4-G). Even though we are using just one of the use cases for our example, you should remember that to create a complete structural model all use cases should be used.

Create CRC Cards

The first step Alec and the team was to create the set of CRC cards by performing textual analysis on the use cases. To begin with, Alec chose the Place Order use case (see Figure 4-G). He and his team then used the textual analysis rules (see Figure 5-1) to identify the candidate classes, attributes, operations, and relationships. Using these rules on the Normal Flow of Events, they identified Customer, Search Request, CD, CD List, and Review as candidate classes. They uncovered three different types of search requests: Title Search, Artist Search, and Category Search. By applying the textual analysis rules to the Brief Description, an additional candidate class was discovered: Order. By reviewing the verbs contained in this use case, they saw that a Customer places an Order and that a Customer makes a Search Request.

To be as thorough as possible, Alec and his team also reviewed the original requirements used to create the use case. The original requirements are contained in Figure 3-A. After reviewing this information, they identified a set of attributes for the Customer (name, address, e-mail, and credit card) and Order (CDs to purchase and quantity) classes and uncovered additional candidate classes: CD Categories and Credit Card Center. Furthermore, they realized that the Category Search class used the CD Categories class. Finally, they also identified three subclasses of CD Categories: Rock, Jazz, and Classical. Alec's goal, at this point in time, was to be as complete as possible. As such, he realized that they may have identified many candidate classes, attributes, operations, and relationships that may not be included in the final structural model. Regardless, the current list of candidate classes included: Customer, Order, Search Request, Title Search, Artist Search, Category Search, CD, CD List, Review, CD Categories, Rock, Jazz, Classical, and Credit Card Center.

Review CRC Cards

The second step for Alec and his team was to carefully review the current set of CRC cards to determine if they had missed any potential candidate classes, attributes, operations, or relationships. Furthermore, the team then used the current candidate classes and a common object list as input to brainstorming additional candidate classes, attributes, operations, and relationships. For example, Alec asked the team members to take a minute and think about what information they would like to keep about CDs. The information that they thought of was a set of attributes, for example, title, artist, quantity on hand, price, and category.

He then asked them to take another minute and think about the information that they should store about orders and an order's responsibilities. The responsibilities they identified were a set of operations, including calculate tax, calculate extension price, calculate shipping, and calculate total. Currently, the attributes (CDs to purchase and quantity) of Order implied that a customer should be allowed to order multiple copies of the same CD and allow different CDs to be ordered on the same order. However, the current structural model did not allow this. As such, they created a new class that was

Front:

Class Name: Customer	**ID:** 1	**Type:** Concrete, Domain
Description: An individual that may or has purchased merchandise from the CD Selections Internet sales system		**Associated Use Cases:** 3

Responsibilities	Collaborators

Back:

Attributes:

First name	State
Middle initial	Country
Last name	Zip code
Street address	E-mail
City	Credit card

Relationships:

Generalization (a-kind-of):

Aggregation (has-parts):

Other Associations: Order; Search Request

FIGURE 5-A Customer Class CRC Card

associated with both the Order class and the CD class: Order Line Item. This new class only had one attribute, quantity, but it had two relationships: one with Order and the other with CD.

When they reviewed the Customer class, they decided that the name and address attributes needed to be expanded; name should become last name, first name, and middle initial, and address should become street address, city, state, country, and zip code. The updated Customer class and Order class CRC cards are shown in Figures 5-A and 5-B, respectively. Once they completed the CRC cards for the Customer and Order class, the team completed the CRC cards for the remaining candidate classes: Order Line Item, Search Request, Title Search, Artist Search, Category Search, CD, CD List, Review, CD Categories, Rock, Jazz, Classical, and Credit Card Center.

Front:

Class Name: Order	**ID:** 2	**Type:** Concrete, Domain
Description: An order that has been placed by a customer which includes the individual items purchased by the customer		**Associated Use Cases:** 3

Responsibilities	Collaborators
Calculate tax	
Calculate shipping	
Calculate total	

Back:

Attributes:

Tax	
Shipping	
Total	

Relationships:

Generalization (a-kind-of):

Aggregation (has-parts):

Other Associations: Order Item; Customer

FIGURE 5-B
Order Class CRC card

Role Play the CRC Cards

The third step was to role-play the classes recorded on the CRC cards. The purpose of this step was to validate the current state of the evolving structural model. Alec handed out the CRC cards to different members of his team. Using the CRC cards, they began executing the different use cases (see Figures 4-B, 4-E, and 4-G), one at a time, to see if the current structural model could support each use case or whether the use case caused the "system" to crash. Anytime the "system" crashed, there was something missing: a class, an attribute, a relationship, or an operation. They would then add the missing information to the structural model and try executing the use case again.

First, Alec and the team decided that the customer had requested the system to perform a search for all of the CDs associated with a specific artist. Based on the current CRC cards, the team felt that the system would produce an accurate list of CDs. They then tried to ask the system for a set of reviews of the CD. At this point in the exercise, the system

crashed. The CRC cards did not have the Review class associated with the CD class. Therefore, there was no way to retrieve the requested information. This observation raised another question. Was there other marketing information that should be made available to the customer, for example, artist information and sample clips?

Next, the team realized that vendor information should be a separate class that was associated with a CD rather than an additional attribute of a CD. This was because vendors had additional information and operations themselves. If the team had modeled the vendor information as an attribute of CD, then the additional information and operations would have been lost. They continued role-playing each of the use cases until they were comfortable with the structural model's ability to support each and every one. Based on the above, the team created CRC cards for the newly identified classes: Mkt Info, Artist Info, Sample Clip, and Vendor.

Create Class Diagram

The fourth step was to create the class diagram from the CRC cards. Figure 5-C shows the current state of the evolving structural model as depicted in a class diagram based on the Places Order use case.

Review Class Diagram

The fifth step was to review the structural model for missing and/or unnecessary classes, attributes, operations, and relationships. At this point, the team challenged all components of the model (class, attribute, relationship, or operation) that did not seem to be adding anything useful to the model. If a component could not be justified sufficiently, then they removed it from the structural model. By carefully reviewing the current state of the structural model, they were able to challenge over a third of the classes contained in the class diagram (see Figure 5-C). It seemed that the CD categories, and their subclasses, were not really necessary. There were no attributes or operations for these classes. As such, the idea of CD categories was modeled as an attribute of a CD. The category attribute for the CD class was previously uncovered during the brainstorming step. Also, upon further review of the Search Request class and its subclasses, it was decided that the subclasses were really nothing more than a set of operations of the Search Request class. This was an example of process decomposition creeping into the modeling process. From an object-oriented perspective, we must always be careful to not allow this to occur. However, during the previous steps in the modeling process, Alec wanted to include as much information as possible in the model. He felt that it was more beneficial to remove this type of information after it had crept into the model than to take a chance on not capturing the information required to solve the problem.

Incorporate Patterns

The sixth step was to incorporate any useful patterns into the structural model (see Figure 5-5). Two patterns that could be useful in this case are the Transaction pattern and the Party pattern (see Figure 5-3). Using these two patterns, Alec and his team uncovered two subclasses of the Customer class, Person and Organization, and they raised the issue of the Place class. However, in this case, all transactions would be taking place in the same place, cyberspace, and therefore the team decided not to include it.

Review the Model

The seventh and final step was to carefully review the structural model. In this case, Alec and the team had to make sure that the CRC cards and the class diagram were in total agreement. Figure 5-D shows the Places Order use case view of the structural model as portrayed in a class diagram developed by Alec and his team. This version of the class diagram incorporates all of the modification described previously.

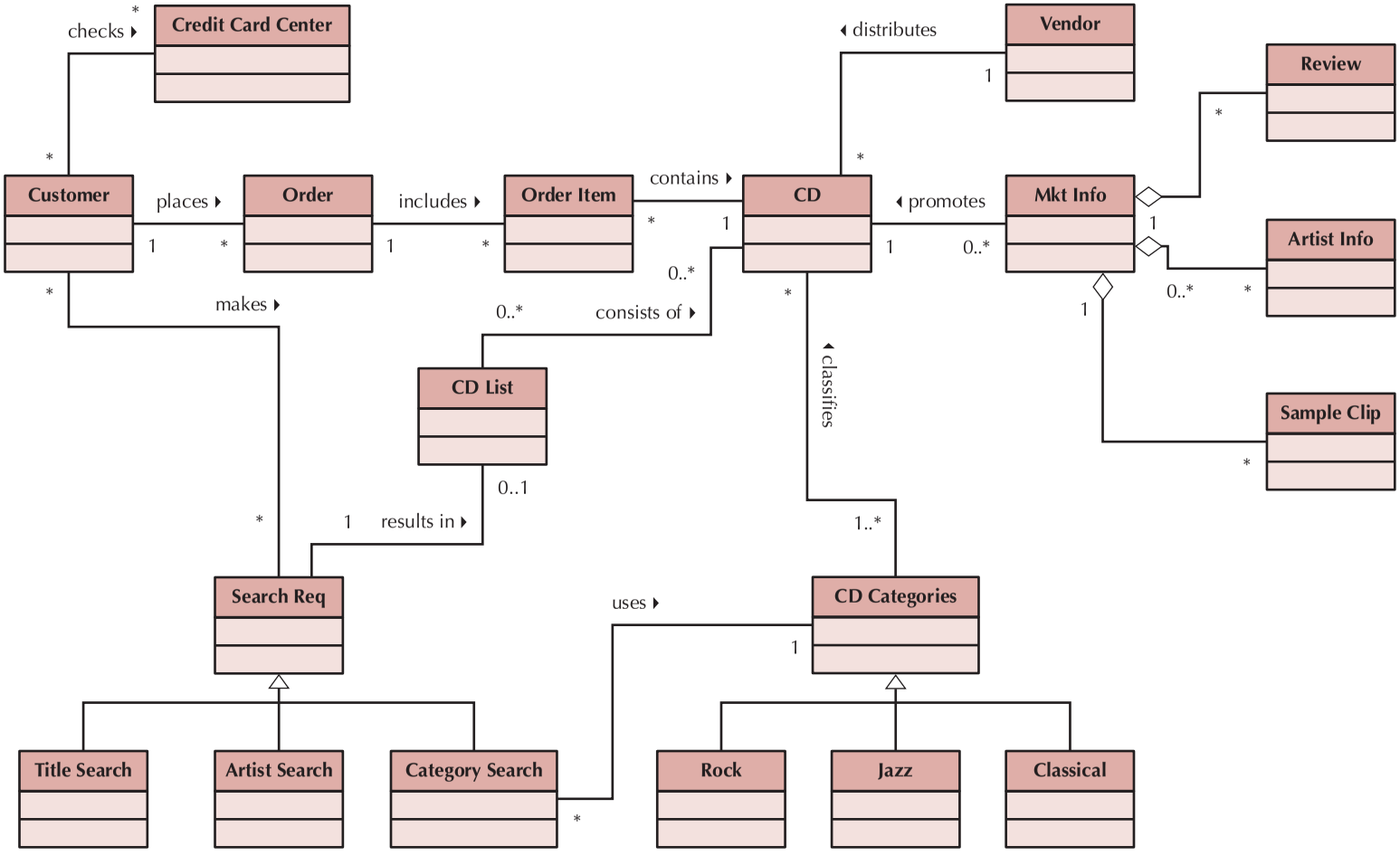

FIGURE 5-C Preliminary CD Selections Internet Sales System Class Diagram (Places Order Use Case View)

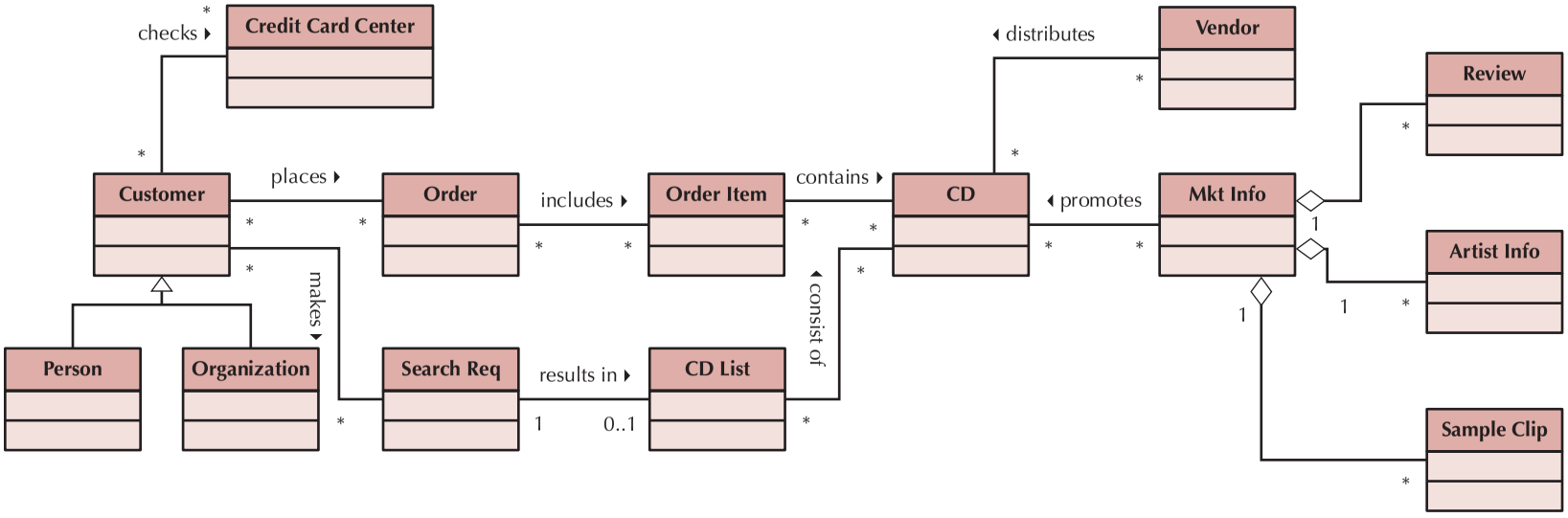

FIGURE 5-D CD Selections Internet Sales System Class Diagram (Places Order Use Case View)