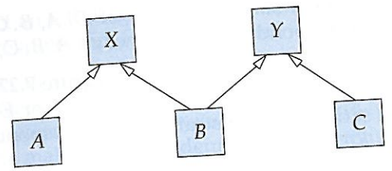

Figure 7.28 E-R diagram for Practice Exercise 7.12.

are inherited from the higher-level entity sets X and Y. Discuss how to handle a case where an attribute of X has the same name as some attribute of Y.

7.13 **Temporal changes**: An E-R diagram usually models the state of an enterprise at a point in time. Suppose we wish to track *temporal changes*, that is, changes to data over time. For example, Zhang may have been a student between 1 September 2005 31 May 2009, while Shankar may have had instructor Einstein as advisor from 31 May 2008 to 5 December 2008, and again from 1 June 2009 to 5 January 2010. Similarly, attribute values of an entity or relationship, such as *title* and *credits* of *course*, *salary*, or even *name* of *instructor*, and *tot_cred* of *student*, can change over time.

One way to model temporal changes is as follows. We define a new data type called **valid_time**, which is a time-interval, or a set of time-intervals. We then associate a *valid_time* attribute with each entity and relationship, recording the time periods during which the entity or relationship is valid. The end-time of an interval can be infinity; for example, if Shankar became a student on 2 September 2008, and is still a student, we can represent the end-time of the *valid_time* interval as infinity for the Shankar entity. Similarly, we model attributes that can change over time as a set of values, each with its own *valid_time*.

a. Draw an E-R diagram with the *student* and *instructor* entities, and the *advisor* relationship, with the above extensions to track temporal changes.

b. Convert the above E-R diagram into a set of relations.

It should be clear that the set of relations generated above is rather complex, leading to difficulties in tasks such as writing queries in SQL. An alternative approach, which is used more widely is to ignore temporal changes when designing the E-R model (in particular, temporal changes to attribute values), and to modify the relations generated from the E-R model to track temporal changes, as discussed later in Section 8.9.

Exercises

7.14 Explain the distinctions among the terms primary key, candidate key, and superkey.

7.15 Construct an E-R diagram for a hospital with a set of patients and a set of medical doctors. Associate with each patient a log of the various tests and examinations conducted.

7.16 Construct appropriate relation schemas for each of the E-R diagrams in Practice Exercises 7.1 to 7.3.

7.17 Extend the E-R diagram of Practice Exercise 7.3 to track the same information for all teams in a league.

7.18 Explain the difference between a weak and a strong entity set.

7.19 We can convert any weak entity set to a strong entity set by simply adding appropriate attributes. Why, then, do we have weak entity sets?

7.20 Consider the E-R diagram in Figure 7.29, which models an online bookstore.

a. List the entity sets and their primary keys.

b. Suppose the bookstore adds Blu-ray discs and downloadable video to its collection. The same item may be present in one or both formats, with differing prices. Extend the E-R diagram to model this addition, ignoring the effect on shopping baskets.

c. Now extend the E-R diagram, using generalization, to model the case where a shopping basket may contain any combination of books, Blu-ray discs, or downloadable video.

7.21 Design a database for an automobile company to provide to its dealers to assist them in maintaining customer records and dealer inventory and to assist sales staff in ordering cars.

Each vehicle is identified by a vehicle identification number (VIN). Each individual vehicle is a particular model of a particular brand offered by the company (e.g., the XF is a model of the car brand Jaguar of Tata Motors). Each model can be offered with a variety of options, but an individual car may have only some (or none) of the available options. The database needs to store information about models, brands, and options, as well as information about individual dealers, customers, and cars.

Your design should include an E-R diagram, a set of relational schemas, and a list of constraints, including primary-key and foreign-key constraints.

7.22 Design a database for a world-wide package delivery company (e.g., DHL or FedEx). The database must be able to keep track of customers (who ship items) and customers (who receive items); some customers may do both.