



Figure 4.12 Aggregation relationship: the classes *Wheel*, *Door* and *Engine* are part of the *Car* class

relationship, using the same notation as for association. The whole end is always assumed to be one, a wheel object, for example, is only part of one car. In Figure 4.12, one car has four wheels, two, four or five doors, and one engine.

In the days before the UML was published, there was much heated debate about the importance and the meaning of aggregation. The UML includes aggregation, but without a precise definition. There is little to distinguish an aggregation relationship from an association relationship. Since this is the case, we feel that aggregation often adds little to the meaning of a model and does not have to be included. The UML includes a stronger form of aggregation, known as composition. Composition is useful because it does have a precisely defined meaning; it is discussed in the Technical points section of this chapter.

*Inheritance and generalization.* If we notice, while modelling classes, that some of them share some common attributes and operations, it can be useful to introduce a new class for the shared bits, leaving only the distinguishing features in the original classes. This process is known as *generalization*. For example, in Figure 4.13 we have two classes we might find in a system for an art gallery: *Photograph* and *Painting*. These two classes both have the attributes *title* and *price* and the operation *updatePrice()*.

We can create a new general class, *Picture*, in which we can place these common features; this is shown in Figure 4.14. *Photograph* and *Painting* retain their distinctive features and share those of *Picture*.

In the structure in Figure 4.14, the classes *Photograph* and *Painting* are *specializations* of the general class, *Picture*. The relationship between a general class and its specializations is known as an *inheritance* relationship. The inheritance mechanism allows the specialized classes to share or inherit the features of the general class. The UML notation for an inheritance relationship is an open-headed arrow which points from the specialized class to the general class.