Polymorphism and Dynamic Binding

An **abstract** method *open* is defined in the parent class FileClass, and the specific implementation of the method appears in each of the three subclasses DiskFileClass, TapeFileClass, DisketteFileClass, and each is given the identical name *open*. At run-time the system determines whether object myFile is an instance of class DiskFileClass, class TapeFileClass, or class DisketteFileClass and automatically invokes the correct method. Since the method *open* can be applied to different classes, it is called **polymorphic** (of different shapes).

Polymorphism enables an object to behalf differently at run time, thus leading to flexible behavior. However, polymorphism and dynamic binding causes difficulty in determining the cause of a failure and leads to maintenance problems.