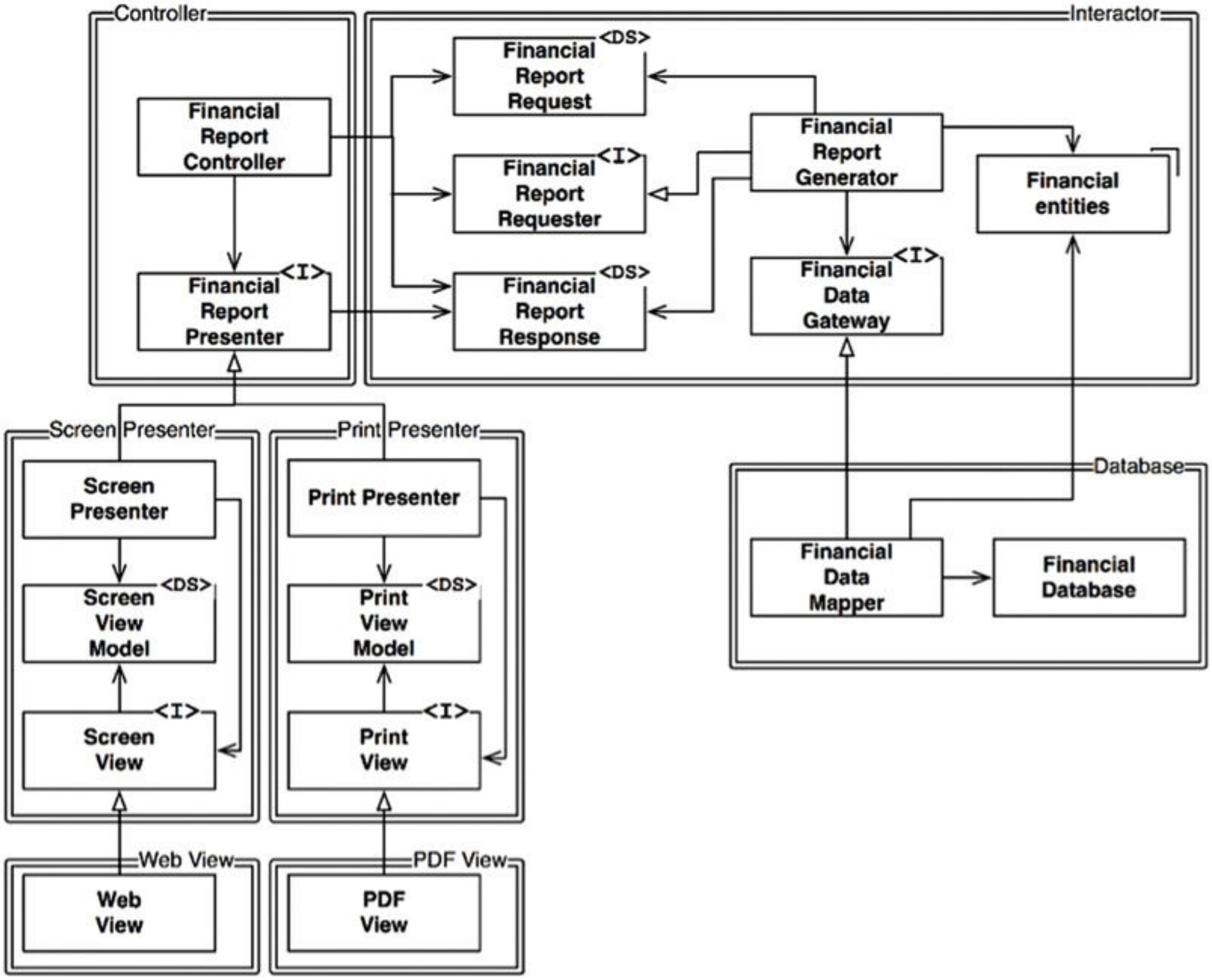

Figure 8.2 Partitioning the processes into classes and separating the classes into components

Classes marked with `<I>` are interfaces; those marked with `<DS>` are data structures. Open arrowheads are *using* relationships. Closed arrowheads are *implements* or *inheritance* relationships.

The first thing to notice is that all the dependencies are *source code* dependencies. An arrow pointing from class A to class B means that the source code of class A mentions the name of class B, but class B mentions nothing about class A. Thus, in Figure 8.2, `FinancialDataMapper` knows about `FinancialDataGateway` through an *implements* relationship, but `FinancialGateway` knows nothing at all about `FinancialDataMapper`.

The next thing to notice is that each double line is crossed *in one direction only*. This means that all component relationships are unidirectional, as shown in the