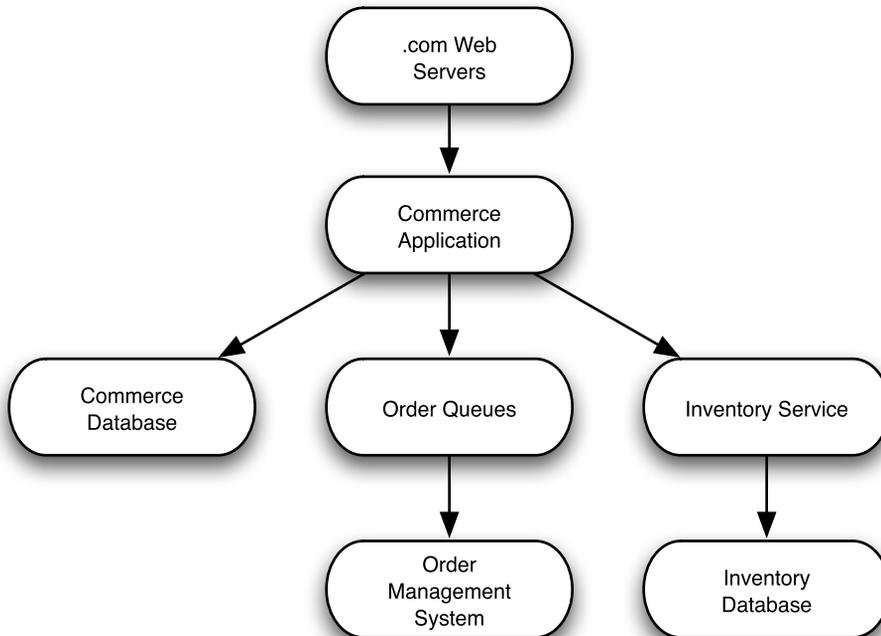


4.3 Cascading Failures



The standard system architecture for enterprise systems, including websites and web services, comprises a collection of functionally distinct farms or clusters that are interconnected through some form of load balancing. We usually refer to the individual farms as *layers*—for example, as in the following graphic—even though they might not really be a single stack.



In a service-oriented architecture, these look even less like traditional layers and more like a directed, acyclic graph.

System failures start with a crack. That crack comes from some fundamental problem. Various mechanisms can retard or stop the crack, which are the topics of the next chapter. Absent those mechanisms, the crack can progress and even be amplified by some structural problems. A cascading failure occurs when a crack in one layer triggers a crack in a calling layer.

A cascading failure occurs when problems in one layer cause problems in callers.

An obvious example is a database failure. If an entire database cluster goes dark, then any application that calls the database is going to experience problems of some kind. If it handles the problems badly, then the application layer will start to fail. One system I saw would tear down any JDBC connection that ever threw a `SQLException`. Each page