2.1.1.2 Incremental development model

The V-model was developed to address some of the problems experienced using the traditional waterfall approach. Defects were being found too late in the life cycle, as testing was not involved until the end of the project. Testing also added lead time due to its late involvement. The V-model provides guidance that testing needs to begin as early as possible in the life cycle, which, as we've seen in Chapter 1, is one of the fundamental principles of structured testing. It also shows that testing is not only an execution based activity. There are a variety of activities that need to be performed before the end of the coding phase. These activities should be carried out in parallel with development activities, and testers need to work with developers and business analysts so they can perform these activities and tasks and produce a set of test deliverables. The work products produced by the developers and business analysts during development are the basis of testing in one or more levels. By starting test design early, defects are often found in the test basis documents. A good practice is to have testers involved even earlier, during the review of the (draft) test basis documents. The V-model is a model that illustrates how testing activities (verification and validation) can be integrated into each phase of the life cycle. Within the V-model, validation testing takes place especially during the early stages, e.g. reviewing the user requirements, and late in the life cycle, e.g. during user acceptance testing.

Although variants of the V-model exist, a common type of V-model uses four test levels. The four test levels used, each with their own objectives, are:

• component testing: searches for defects in and verifies the functioning of software components (e.g. modules, programs, objects, classes etc.) that are separately testable;

• integration testing: tests interfaces between components, interactions to dif ferent parts of a system such as an operating system, file system and hard ware or interfaces between systems;

• system testing: concerned with the behavior of the whole system/product as defined by the scope of a development project or product. The main focus of system testing is verification against specified requirements;

• acceptance testing: validation testing with respect to user needs, require ments, and business processes conducted to determine whether or not to accept the system. The various test levels are explained.

In practice, a V-model may have more, fewer or different levels of development and testing, depending on the project and the software product. For example, there may be component integration testing after component testing and system integration testing after system testing. Test levels can be combined or reorganized depending on the nature of the project or the system architecture. For the integration of a commercial off-the-shelf (COTS) software product into a system, a purchaser may perform only integration testing at the system level (e.g. integration to the infrastructure and other systems) and at a later stage acceptance testing.

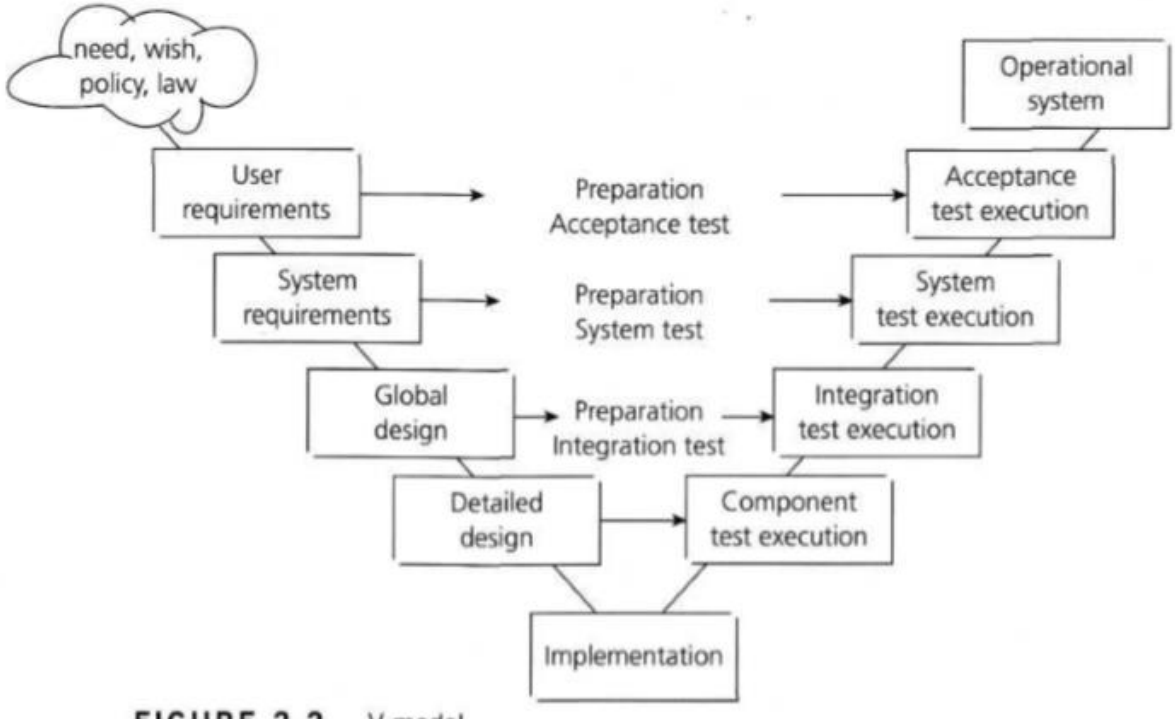

FIGURE 2.2 V-model