

What is the difference between verification and validation?
Which types of testing address verification? Which types of testing address validation?

Raoul Jetley and Ben Chelf summarize the FDA's definition of verification and validation:

Verification and validation are terms that are often used in software. However, it is important to understand the difference between these two distinct but complementary activities. Software verification provides objective evidence that the design outputs of a particular phase of the software development life cycle meet all of the specified requirements for that phase by checking for consistency, completeness, and correctness of the software and its supporting documentation. Validation, on the other hand, is the confirmation by examination and provision of objective evidence that software specifications conform to user needs and intended uses, and that the particular requirements implemented through software can be consistently fulfilled (Raoul & Chelf, 2009).

As verification focuses on the correctness of the system, its outputs and consistency in the context of the designers' understanding of the requirements specifications, function and performance tests are best for verification testing. The function test takes the integrated system and checks that it performs all functions described in the functional requirements properly. The performance test checks the system against the nonfunctional requirements, that the system is fast enough, secure enough, and reliable enough to satisfy these requirements.

Since validation focuses on ensuring the system actually satisfies the users needs and intentions, acceptance and installation tests verify the working system meets these requirements. The acceptance test allows the customer to validate that the system built according to the requirements is actually what they need. The installation test allows the user to test-drive the system in the environment where it will actually be used, further validating that it will work as needed (Pfleeger & Atlee, 2006).

References:

Jetley, Raoul and Chelf, Ben (2009). *Diagnosing Medical Device Software Defects Using Static Analysis*. Medical Device & Diagnostic Industry. Retrieved from devicelink May 30, 2009: <http://www.devicelink.com/mddi/archive/09/05/006.html>

Pfleeger, S., & Atlee, J. (2006). *Software Engineering: Theory and Practice*. New Jersey: Prentice Hall.