V&V Summary

V&V Definitions

Verification

Verification is the process of determining whether a system completely satisfy its specifications. In other words, verification is checking that the system is built right. It focuses on the matching between the system and the stated specifications.

Validation

Validation is the process of determining whether a system satisfactorily perform the real-world tasks for which it was created. Another definition of Validation states that it is the process of determining if a system satisfy the explicit or implicit needs of the user. In any case, validation is checking that the right system has been built. Clearly, this concept is based on the implicit assumption that, despite the fact that detailed and accurate system specifications are provided, what the user (or the organization) actually want lies only in the heads of the people involved. This can hardly be clearly expressed in verbal or formal terms, so the ultimate check is only to put the system into practice.

Scope of V&V

From another perspective, we can differentiate validation from verification with respect to the level of abstraction (or granularity) of the elements involved. Validation is used at system or subsystem level, whereas Verification is performed at a more variable granularity: from a single line of source code, to single units or classes, and the from cluster of units (modules and subsystem) to the entire system.

Activities of V&V

Verification

Software inspection (of documentation, specifications, design, and source code).
    Mode: Static
Testing (of specified functionalities)
    Mode: Dynamic

Validation

Validation testing (direct use of the system)
    Mode: Dynamic
Prototyping
    Mode: Dynamic
Inspection of Requirements (Requirement validation)
    Mode: Static