Project Risk Management:
IV&V as Insurance for Project Success

Introduction

Software development projects can be expensive and risky: Ever more complex mission-critical requirements lead to increasingly more complicated software systems. Now add to that the fact that the cost to correct software errors multiplies during the software development life cycle (SDLC).

Independent verification and validation (IV&V) services establish confidence among program sponsors and users that a software project under development will meet all of its requirements and that it will be deployed on time and within budget. IV&V is a proven methodology that mitigates project risk by identifying errors that can lead to time and cost overruns and even catastrophic system failure at the time of deployment.

The Institute for Electrical and Electronics Engineers (IEEE) standard 1012-2004, *Standard for Software Verification and Validation*, defines *verification* by asking, “Is the system built right?” Verification determines whether the products of a given SDLC phase fulfill established requirements.

The same IEEE standard defines *validation* by asking, “Did the right system get built?” Validation evaluates the software at the end of the SDLC to ensure that it complies with all user requirements and performs as expected.

IV&V should not be perceived as an adversarial process—it is not about a lack of confidence in the software developers. Instead, it is akin to taking out an insurance policy: Investing in a financially and managerially independent organization to monitor all phases of software development and testing ensures that errors and anomalies are detected and corrected early in the process, which in turn reduces costs and saves time.

IV&V Defined

IV&V is formally defined as an engineering discipline that employs rigorous methods for evaluating the correctness and quality of the software product throughout the software development life cycle from a *system-level* point of view. Rather than being an integral part of the SDLC, it takes place side by side with software development, testing, and integration.

The key point in the IV&V concept is *independence*, which is defined as follows:

- *Technical independence*—IV&V personnel use their expertise to assess development processes and products independently of the developer.
- *Managerial independence*
  - The IV&V effort is vested in an organization that is separate from the organization responsible for system development.
  - The IV&V team independently selects the software components to be analyzed and the IV&V techniques to be used.

  - The IV&V team independently defines the schedule of IV&V activities.
- *Financial independence*—All IV&V work is funded separately from the development work.

QC and QA versus IV&V

Quality control (QC) occurs *as part of* the software development team's tasks—the same vendor or contractor who designs, develops, tests, and implements the system also performs the QC functions on the software. And because the same vendor or contractor is performing the QC functions, there is no one who can offer outside, "fresh eyes" oversight during the project.

Another method of ensuring the quality of a software project is quality assurance (QA). The IEEE *Handbook of Software Quality Assurance* provides the following definition: "Software quality assurance is the set of systematic activities providing evidence of the ability of the software process to produce a software product that is fit to use."

At first glance, this looks similar to the goals of IV&V. The primary difference, however, is that even independent QA efforts usually fall within the scope of the project management office. This situation means that the QA functions are not managerially independent—a key feature of IV&V.

The issue that development and some QA teams face is that their members become deeply involved in the intricate, day-to-day challenges of the project, which, as part of human nature, makes it extremely difficult to pull back from the trees and really see the whole forest. That is the job of the IV&V team.

Characteristics of IV&V

Scalability

In addition to its signature independence, the IV&V process is *scalable* and *flexible*—it is not a rigid methodology, but rather, one that can be tailored to any size project to address the unique technical, functional, budgetary, and performance requirements in terms of the project's size and complexity. A more complex project or system will have more phases that it must pass through, and thus there will be more critical junctures at which IV&V analysis should be employed—compare, for example, a major upgrade to financial systems modules in a large Cabinet-level federal agency to a new employment application tracking program at a small private business. IV&V can be used for both, but at radically different levels of effort.

Criticality Analysis

Criticality analysis is one distinguishing characteristic of effective IV&V services. The purpose of criticality analysis is to focus scarce IV&V resources on the most crucial parts of the SDLC.

Criticality analysis is a structured process that correlates system hazards to system components, establishing the integrity levels necessary to focus attention on the vital aspects of the project. It applies and reexamines software integrity levels of system components at the beginning of each IV&V process, as well as the effect that the failure of each component would have on overall system effectiveness.

This analysis provides the basis for

- focusing the attention of development and testing resources on the key system components,
- applying limited IV&V resources where they are needed most,
- identifying measures to mitigate failures and support contingency and business continuity planning, and
- detailing the criteria for final system certification.

Is IV&V Just a Trend?

Independent verification and validation is an established, proven process. IEEE standard 1012-2004, *Standard for Software Verification and Validation*, is the industry-wide gold standard for including IV&V in software development projects.

In the late 1950s, software began to be included in systems procured by the U.S. Department of Defense (DoD). These projects were consistently behind schedule and over budget, and had many technical problems. Frequently, software did not work as intended and many projects were canceled before any deliverables could be met. Throughout this period, software development contractors often gave overly optimistic assessments of the software development status to DoD. DoD was frequently unaware of schedule, budget, and technical problems until late into the program—when they were often unable to understand them and assess their effects.

The IV&V role was established to address this problem. The first program to use IV&V was the Atlas Missile Program in the late 1950s. An independent software tester was hired to perform additional, unbiased software testing. By employing an outside entity that was completely separate from the software development contractor, DoD hoped to get a more accurate and objective technical assessment of a project's status.

Over time, the role of the "IV&V contractor" became critically important. IV&V has been and still is used on many large, mission-critical projects for DoD, NASA, the Federal Aviation Administration, the Department of Housing and Urban Development, and the Drug Enforcement Agency.

The increasing complexity, size, and importance of software applications has led to a steadily increasing demand for IV&V.

Implementing IV&V

Software project sponsors should consider establishing the IV&V process as early as the project planning phase so that the IV&V team members will have an unbiased, impartial view into project planning, scheduling, budgeting, and resource allocation.

Including the IV&V team at the beginning of the project ensures development team compliance with the mandated scope and functionality of the software. Early implementation of IV&V helps prevent cost overruns and schedule slippages by means of both preventive and corrective actions to correct risks, gaps, and issues detected in early phases of the SDLC. This approach serves to increase developer productivity and customer satisfaction.

IV&V activities are performed concurrently with each SDLC phase to provide timely feedback to the software development team so that its members can develop and implement a quality software product. The major IV&V phases include

- requirements verification (requirements validation and requirements traceability matrix evaluation),
- design verification (requirements traceability and sufficiency of design),
- development (code verification and software unit/component testing),
- system validation (functional, performance, technical, conversion, and data-cleansing validation),
- system implementation (readiness for production roll-out), and
- operations and support (requirements management and system modifications).

SDLC in Brief

As its name implies, the software development life cycle is a continuous process, because any given software application is bound to require updates and maintenance during its lifetime.

As Figure 1 below shows, a new software development project starts with requirements definition—a comprehensive survey and analysis of the functionality that the final software system must incorporate to satisfy users' needs. This functionality is usually documented in a requirements definition document.

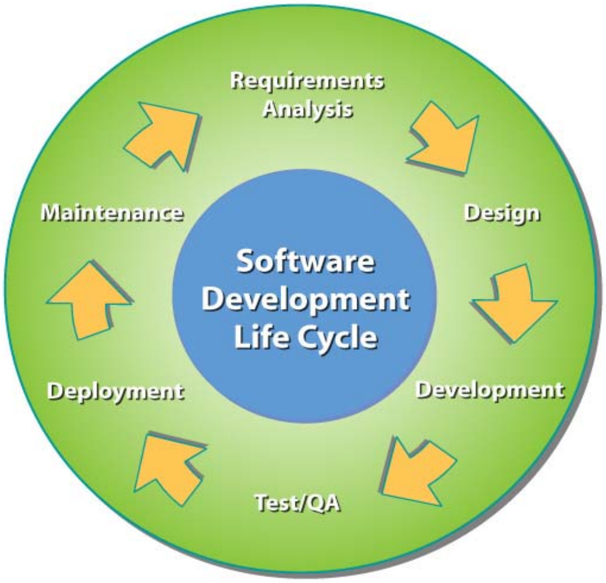

Figure 1. Software Development Life Cycle

The next step is to design the system based on the identified requirements. The design phase may include several increments, each one drilling down deeper into the details of each of the modules that comprise the system. A detailed design document is usually compiled at this stage, for use as a guide for the system developers.

The development stage is the one during which the code is written. For a system with more than one module, each module is coded individually.

During the test/QC phase, individual modules are tested (*unit testing*). Once unit testing is complete and each module is shown to function as intended, the modules are combined into the complete system and tested (*integration testing*). The goal of integration testing is to ensure that there are no conflicts in the way the modules interact with each other. *System testing* shows whether the complete system behaves as expected. *Acceptance testing* is the final step to ensure that the system is ready to be placed into production for everyday use. Test cases are the most common tool for all types of testing, and detailed logs should be maintained to document all test results and any redesign or redevelopment effort that is required to fix errors.

Deployment is the stage at which the fully tested system is moved into production—that is, it becomes available for the end users.

Finally, any system is likely to need maintenance, which can include upgrades to existing functionality, creation of new reporting tools, and minor bug fixes.

The Role of IV&V alongside SDLC

At the heart of a successful IV&V methodology is the verification of each phase of the SDLC relative to itself and adjacent phases.

As Figure 2 shows, IV&V activities take place separate from, but in concert with, the SDLC stages, giving the IV&V team the chance to assess future risks and suggest redirection at each stage in the process. Each IV&V phase is designed to determine whether development products of an SDLC activity conform to the requirements defined for that activity and whether the software satisfies its intended use.

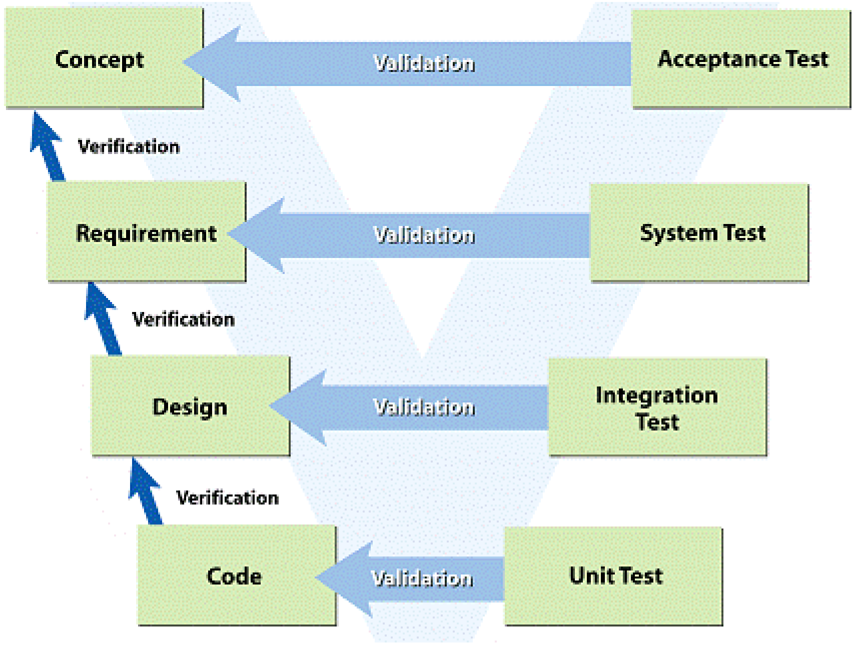

Figure 2. Primary IV&V Activities

IV&V Phases

A well-conceived IV&V methodology mirrors the development process, during which requirements are fully defined before the design is created; the top-level and detailed designs fully implement the approved requirements before development/coding; and development/coding is tested at each level of component build and integration before final system validation and acceptance testing.

Planning Phase

The IV&V team reviews all the project deliverables—including project plans and schedules, costs, the work breakdown structure (WBS), the configuration/change management (CM) plan, the quality control/assurance (QC/QA) plan, and data management plans—to determine whether they are realistic and can be implemented within the available resources and constraints. The team screens and cross-references all project deliverables against preliminary system requirements documents to detect such things as issues, shortfalls, errors, and discrepancies.

The following table contains some sample questions from an IV&V checklist that would be used to monitor the planning phase.

Criteria	Questions
Planning	▪ Is the Project Statement—scope, definition, and objectives—the same as agreed to in the Project Statement process? ▪ Has the Project Statement been reviewed as part of the baseline process?
Other Plans (as needed)	▪ Is there a Documentation Plan? ▪ Is there a Training (Knowledge Transfer) Plan?

Criteria	Questions
Other	■ Is the Development Methodology well defined?
Other Planning Criteria	■ Is the plan for project resources adequate? ■ Are the original project schedule and budget realistic?
Organization	■ Is the project organization documented and on file? ■ Have team roles and responsibilities been documented and clearly communicated to the team, client, and stakeholders?
Communications	■ Is there a written project communications plan? ■ Are all stakeholder groups and key stakeholders identified?
Tracking & Monitoring	■ Have the following project metrics been defined and methods devised for their collections, analysis, and reporting: o milestone dates? o project schedule?
Reviewing	■ Have the deliverable review, evaluation, and acceptance criteria been established? ■ Have the various responsibilities and frequencies for deliverable reviews been defined and communicated?
Issue Management	■ Is an Issue Management Process documented? ■ Has this process been communicated to the client and team members?
Change Control	■ Will a Change Control Process be in place? ■ If so, is the Change Control Process documented and on file?

Requirements Analysis Phase

The IV&V team should participate proactively in any requirements elicitation sessions to meet with the end users to understand their business needs, taking notes as needed. When the Functional Requirements Document (FRD) is finalized, the IV&V team reviews it and compares it with the notes to ensure that the system requirements written in the FRD cover all of the users' needs and are implementable as well as testable.

The following table contains some sample questions from an IV&V checklist that would be used to monitor the requirements phase.

Criteria	Questions
Compatibility	■ Do the interface requirements allow compatibility of external interfaces (hardware and software)?
Completeness	■ Do the requirements specifications contain everything listed in the corresponding documentation content? Does it include all requirements relating to functionality, performance, constraints, safety, etc.? ■ Are the environmental conditions specified for all operating modes (for example, normal, abnormal, disturbed)?
Consistency	■ Is there internal consistency between the software requirements? ■ Are the requirements specifications free of contradictions?
Correctness	■ Do the requirements specifications conform to standards? ■ Are algorithms and regulations supported by appropriate literature?
Feasibility	■ Will the design, operation, and maintenance of software be feasible?

Criteria	Questions
	▪ Are the specified models, numerical techniques, and algorithms appropriate for the problem to be solved? Can they be implemented within the imposed constraints?
Modifiability	▪ Are requirements organized so as to allow for modification? ▪ Are there any redundant statements?
Robustness	▪ Are there requirements for fault tolerance and graceful degradation?
Traceability	▪ Is there traceability from the next higher-level specification? ▪ Do the requirements specifications show explicitly the mapping and complete coverage of all relevant requirements and design constraints, by such means as a coverage matrix or cross-reference?
Understandability	▪ Does every requirement have only one interpretation? ▪ Are the functional requirements in modular form, with each function explicitly identified?
Verifiability & Testability	▪ Are the requirements verifiable (that is, can the software be checked to see whether requirements have been fulfilled)? ▪ Is there a verification procedure defined for each requirement in the requirements specifications?

Design Phase

The IV&V team reviews the system design document (SDD) to determine whether all the system physical and logical designs are implementable as well as testable.

The following table contains some sample questions from an IV&V checklist that would be used to monitor the design phase.

Criteria	Questions
Completeness	▪ Are all the items listed in requirements addressed in the design documentation? ▪ Are the requirements fulfilled?
Consistency	▪ Are standard terminology and definitions used throughout the design documentation? ▪ Are the style of presentation and the level of detail consistent throughout the design?
Correctness	▪ Does the design documentation conform to design documentation standards? ▪ Does the design implement only that which is specified and approved in the requirements specifications, unless additional functionality is justified?
Feasibility	▪ Are the specified models, algorithms, and numerical techniques accepted practices for use within this application? ▪ Can they be implemented within the constraints imposed on the system and on the development effort?
Modifiability	▪ Are the modules organized such that changes in the requirements require changes to only a small number of modules? ▪ Do the functions and data structures that are likely to change have interfaces insensitive to changes in individual functions?
Predictability	▪ Does the design contain programs that provide the required response to identified error conditions? ▪ Does the design schedule computer resources in a manner that is primarily deterministic and predictable rather than dynamic?

Criteria	Questions
Robustness	■ Are all approved software requirements that are related to fault tolerance and graceful degradation addressed in the design?
Structure	■ Does the design use a logical hierarchical control structure?
Traceability	■ Does the design documentation show mapping and complete coverage of all approved requirements? ■ Are all functions within the scope of the approved requirements identified?
Understandability	■ Does the design documentation avoid unnecessarily complex designs and design representations? ■ Is the design documentation written to allow unambiguous interpretation?
Verifiability & Testability	■ Does the design documentation describe each function using well-defined notation so that it can be verified against the requirements specifications and so that the code can be verified against the design documentation? ■ Are conditions and constraints identified quantitatively so that tests may be designed?

Development Phase

The IV&V team inspects the source code to ensure that it meets the coding standards, and correctly and completely implements all the specified system requirements.

The following table contains some sample questions from an IV&V checklist that would be used to monitor the development phase.

Criteria	Questions
Completeness	■ Does the source code cover all the requirements specified in the Functional Requirements Documents (FRD)? ■ Does the source code implement all the functionalities described in the System Requirements Specifications (SRS)?
Compatibility	■ Does the source code conform to the coding standards? ■ Are the coding conventions followed?
Consistency	■ Are the coding standards and conventions used? ■ Are the variable names used?
Modifiability	■ Is the program divided into modules for easy maintenance and modification?
Traceability	■ Do all the program modules have their own identification for tracing purposes? ■ Do the program modules reference the requirements and functions being implemented?
Understandability	■ Are comment lines provided for each module to explain which requirements and functions are being implemented in each module?
Correctness	■ Are all specified requirements and functionalities implemented correctly? ■ Are errors being handled correctly in case of functional failures?
Adequacy	■ Is an adequate level of program control provided to avoid dormant code lines? ■ Is an adequate level of program control provided to avoid indefinite looping?

Test Phase

The test phase includes three IV&V activities:

- *System Readiness Assessment*—The IV&V team assesses, via the Test Readiness Review (TRR) meetings, the system readiness for the unit, subsystem, and system integration tests to be performed by the development team. The IV&V team develops TRR checklists to track the progress.
- *Test Monitoring*—The unit, subsystem, and system integration tests performed by the development contractor are monitored by the IV&V team to ensure that
    - these tests are executed in accordance with the approved test plans and procedures,
    - all testable requirements are accounted for and are mapped to corresponding test cases in a requirements traceability matrix (RTM), and
    - all test results, testing status, and anomalies are recorded and documented in a test analysis report (TAR).
- *Independent Testing and Evaluation (ITE)*—The IV&V team performs ITE to verify that all system requirements, such as functional, operational, performance, and interface requirements, are met and to detect system anomalies for correction before the system is deployed. The team develops an ITE test plan to describe all planning steps, testing resources, and timelines. Step-by-step IV&V test cases are also developed and mapped to the system requirements at this time for tracking purposes.

  The team executes the IV&V test cases to obtain the actual output produced by the system under test when the defined input is fed into the system. The actual output is evaluated and compared with the expected output to determine the system behavior. The team also writes up the Test Problem Reports (TPR) when anomalies are detected.

The following table contains some sample questions from an IV&V checklist that would be used to monitor the testing phase.

Criteria	Questions
Completeness	▪ Are all test activities planned and documented? ▪ Have the resources, project team responsibilities, and management activities needed to plan, develop, and implement the testing activities that will occur throughout the life cycle been identified?
Consistency	▪ Have the occurrence and timing of the test phases in the life cycle and the entrance and exit criteria for each test phase been identified? ▪ Have criteria for evaluating the test results of each test phase been established?
Correctness	▪ Do testing materials conform to project standards? ▪ Are discrepancies identified and added to the problem report, and have problems been resolved?
Traceability	▪ Has mapping occurred to identify which requirements are verified in what test phase?
Understandability	▪ Is there a glossary of terms?
Verifiability & Testability	▪ Are system and acceptance testing of the software planned and performed to demonstrate that the software satisfies its requirements? ▪ Has the necessary testing been established to validate that the project requirements have been met?
Accuracy	▪ Has the Test Plan been reviewed with the system owner before any tests have been conducted?

Criteria	Questions
Adequacy	■ Is the testing adequate to verify the functionality of the software product?

Deployment Phase

The IV&V team reviews the implementation and deployment plans for quality assurance purposes and attends any Production Readiness Review (PRR) meetings to determine whether the system is ready to be deployed in accordance with the implementation and deployment plans.

The following table contains some sample questions from an IV&V checklist that would be used to monitor the deployment phase.

Criteria	Questions
Infrastructure and System Readiness	■ Is each infrastructure component in place, fully tested, and accepted by the client as being operationally ready? ■ Are all the software products that are required to correctly operate the system present in the installation package?
System Monitoring and Support Readiness	■ Do the planned system monitoring tools and related process meet the system monitoring requirements of the client? ■ Is the staff that will support the system users adequately trained?
Deployment Management	■ Are the problem identification and resolution tracking processes, including escalation criteria, in place and appropriate for deployment? ■ Are the risks associated with any outstanding issues acceptable to the client?
Operations and Maintenance Readiness	■ Are all key components required for the operation and maintenance of the system tested and in place? ■ Are the technical user manuals, systems maintenance manuals, and/or systems operational manuals reviewed and ready for use?

Conclusion

Independent verification and validation is a time-tested methodology that should be carefully considered as a relatively low-cost insurance policy at the outset of any software development project.

Having an IV&V team working side by side with developers can be critical to the successful development of programs, projects, and products and to the provision of services because its members provide independent oversight throughout the life of the project. This oversight ensures that the developers can meet all program plans, schedules, and deliverables with fewer hidden errors, issues, and anomalies, thus lowering the risk of unexpected cost and schedule overruns.

The IV&V team's findings will be unbiased and truthful because the team is financially, technically, and managerially independent from the developers.

Futrend Technology, Inc.

As a Small Business Administration (SBA) certified 8(a) corporation, Futrend Technology, Inc., has positioned itself to assist federal clients with tasks ranging from systems engineering and development of Enterprise Resource Planning (ERP) solutions, software development, system integration, and implementation, to full life-cycle information technology support and professional services such as IV&V, independent quality assurance, and certification and accreditation (C&A). We focus on maintaining our ability to “partner” with each of our customers to provide top-notch professional services and excellent results.

Futrend prides itself on excellent service delivery. Our corporate leadership focuses on providing quality services that meet or exceed expectations.