Test and Evaluation Strategy Primer

The Test and Evaluation Strategy (TES) is an early Test and Evaluation (T&E) planning document that is required at Milestone A and describes the T&E activities starting with Technology Development and continuing through System Development and Demonstration into Production and Deployment. Over time, the TES scope expands and evolves into the T&E Master Plan (TEMP) at Milestone B. The TES describes, in as much detail as possible, the risk reduction efforts across the range of activities (e.g., M&S, DT&E, OT&E, etc.) that will ultimately produce a valid evaluation of operational effectiveness, suitability, and survivability before full-rate production and deployment. It is a living document and is updated during the Technology Development Phase. Its development requires early involvement of testers, evaluators, and others as a program conducts pre-system acquisition activities, especially prototype testing. The TES should be consistent with and complementary to the Systems Engineering Plan and the TEMP.

The TES contains hardware and software maturity success criteria used to assess key technology maturity for entry into System Development and Demonstration. The TES is the tool used to begin developing the entire program T&E Strategy, and includes the initial T&E concepts for Technology Development, System Development and Demonstration and beyond. For programs following an evolutionary acquisition strategy with more than one developmental increment, the TES should describe how T&E and M&S would be applied to confirm that each increment provides its required operational effectiveness, suitability, and survivability, as would be required of a program containing only one increment. Its development establishes an early consensus on the scope of how the program will be tested and evaluated, with particular consideration given to needed resources, in order to support PPBE process activities.

Program managers are required to develop and fund a T&E Strategy that meets the following objectives:

- Perform verification and validation in the systems engineering process;
- Develop an event-driven T&E Strategy, rather than a schedule-driven one, to ensure program success;
- Identify technological capabilities and limitations of alternative concepts and design options under consideration to support cost-performance tradeoffs;
- Identify and describe design technical risks. The TES should naturally flow from the systems engineering processes of requirements analysis, functional allocation, and design synthesis;
- Stress the system under test to at least the limits of the Operational Mode Summary/Mission Profile, and for some systems, beyond the normal operating limits to ensure the robustness of the design and that expected operational performance environments can be satisfied;
- Assess technical progress and maturity against Critical Technical Parameters (CTPs), including interoperability, documented in the TEMP. As part of an event-driven strategy, the use of success criteria is a suggested technique with which program managers can meet this requirement. Technical parameters, such as levels of reliability growth or software maturity, increasing loading levels, mission processing timelines, and the like, can be used as success criteria to assess technical progress;
- Assess the safety of the system or item to ensure safe operation during OT&E, other troop-supported testing, operational usage, and to support success in meeting design safety criteria;
- Provide data and analytic support to the decision process to certify the system ready for OT&E;
- Conduct information assurance testing on any system that collects, stores, transmits, and processes unclassified or classified information. The extent of IA testing depends upon the assigned Mission Assurance Category and Confidentiality Level. DoD Instruction 8500.2 mandates specific IA Control Measures that a system should implement as part of the development process;
- In the case of IT systems, including NSS, support the DoD Information Technology Security Certification and Accreditation Process and Joint Interoperability Certification process;
- Discover, evaluate, and mitigate potentially adverse electromagnetic environmental effects (E3);
- Support joint interoperability assessments required to certify system-of-systems interoperability;
- In the case of financial management, enterprise resource planning, and mixed financial management systems, the developer shall conduct an independent assessment of compliance factors established by the Office of the USD(C);
- Prior to full-rate production, demonstrate the maturity of the production process through Production Qualification Testing of LRIP assets. The focus of this testing is on the contractor's ability to produce a quality

product, since the design testing should already have finished. Depending on when this testing is conducted, the results might be usable as another data source for IOT&E readiness determinations; and
- Demonstrate performance against threats and their countermeasures as identified in the DIA-validated System Threat Assessment. Any impact on technical performance by these threats should be identified early in technical testing, rather than in operational testing where their presence may have serious repercussions.

In addition to the mandatory items above, the following items are strongly recommended to ensure a robust T&E program:
- Involve testers and evaluators, from within the program and outside, early in T&E planning activities to tap their expertise from similar experiences and begin identifying resource requirements needed for T&E budgeting activities;
- Ensure the T&E Strategy is aligned with and supports the approved acquisition strategy, so that adequate, risk-reducing T&E information is provided to support decision events;
- Utilize developmental test activities, where appropriate, to include hardware-in-the-loop simulation, prior to conducting full-up, system-level testing in realistic environments;
- The required assessment of technical progress should include reliability, desired capabilities, and satisfaction of Critical Operational Issues (COIs) to mitigate technical and manufacturing risks;
- Increase likelihood of OT&E success by testing in the most realistic environment possible;
- Assess system-of-systems Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance (C4ISR) prior to OT&E to ensure that interoperability under loaded conditions will represent stressed OT&E scenarios.

There is no prescribed format for the TES, but it should include the following items, to the extent known:
- Introduction and objectives of the system-specific technical and operational evaluations that will support future decision events;
- System description, mission, concept of operations, and major performance capabilities from the Initial Capabilities Document. Identify new technology and the plan to identify associated risk;
- Acquisition strategy concept - For programs following the preferred evolutionary acquisition strategy, the TES should describe how T&E and M&S would be applied to each increment. It should show how each increment would ultimately provide a demonstrated level of operational effectiveness, suitability, and survivability, and meet user needs with a measurable increase in mission capability;
- Time-phased threats to mission accomplishment;
- Anticipated concept of operations, including supportability concept;
- Technical risk reduction testing, including any new or critical technologies identified in the Technology Development Strategy;
- Anticipated component and sub-system developmental testing that begins after MS A;
- Test and evaluation strategy for System Development and Demonstration;
- Critical operational and live fire (if appropriate) issues;
- Scope and structure of the operational and live fire evaluations;
- Likely sources of required data;
- Major T&E design considerations;
- Hardware and software maturity success criteria;
- T&E schedule;
- Anticipated M&S used for future system evaluations; and
- T&E funding estimates in enough detail to permit programming and budgeting.

DoDD 5000.01 requires T&E to be integrated throughout the defense acquisition process and structured to provide essential information to decision-makers, assess attainment of technical performance parameters, and determine whether systems are operationally effective, suitable, survivable, and safe for intended use. T&E conduct, integrated with modeling and simulation, should facilitate learning, assess technology maturity and interoperability, facilitate integration into fielded forces, and confirm performance against documented capability needs and adversary capabilities as described in the system threat assessment. Integrated T&E is defined as: "the collaborative planning and collaborative execution of test phases and events to provide data in support of independent analysis, evaluation and reporting by all stakeholders particularly the developmental (both contractor and Government) and operational test and evaluation communities." Overviews of the four types of T&E that should be considered for integration are presented in Table 1.

Table 1. T&E Type Overviews

DT&E	OT&E	Interoperability T&E	IA T&E

• Controlled by PM
• One-on-one tests
• Controlled environment
• Contractor involvement
• Trained, experienced operators
• Precise performance objectives and threshold measurement
• Test to specification
• Development test article
• Test conditions determined by the PM
	• Controlled by independent test agency
	• Many-on-many tests
	• Realistic/tactical environment with operational scenario
	• Restricted system contractor involvement
	• User troops recently trained on equipment
	• Performance measurement of operational effectiveness and suitability
	• Test to requirements
	• Production representative test article
	• Test conditions operationally realistic with typical users and logistics support
		• Certification conducted by JITC
		• Operational testing conducted by OTA(s)
		• Includes NR-KPP
		  ○ Adherence to NCOW Reference Model
		  ○ Required Integrated Architecture Products
		  ○ Adherence to Key Interface Profiles (KIPs)
		  ○ Information Assurance
		• Test conditions reflect an applicable capability environment
			• Controlled by OTA, DIA, FSO, NSA
			• System Registration (JCPAT-E and DOD IT Registry)
			• Interconnectivity and Interoperability Capability Profile
			• IT Standards Profile J-6 Interoperability Certification Memorandum Provided to Document Sponsor Via KM/DS
			• Test conditions are a mix of operational and laboratory environments

Integrating T&E consists of many aspects, all designed to optimize test scope and minimize cost. For example, separate contractor developmental testing might be combined with governmental developmental test and evaluation, with control being exercised by a combined test organization. Live testing might be integrated with verified, validated, and accredited simulators or computer driven models and simulations, to optimize the amount of live testing required. Another aspect is integrating DT&E with OT&E into a continuum that reduces testing resource requirements and time, or conducting concurrent DT and OT when objectives and realism are compatible. Another approach is to combine T&E types into a single test event, with data provided to the various T&E evaluators equally. There is no single integration solution that is optimum for all programs, but each program should consider the T&E approaches presented in Table 2 in preparing the T&E strategy.

It is important that the DT&E, OT&E, Interoperability or IA objectives are not compromised when integrating T&E. The user community should be involved early in test planning to ensure the statement of desired capabilities is interpreted correctly and tested realistically. No single T&E integration approach may be right for every program. In some cases, in order to maximize cost and schedule savings, a program may use a combination of integration approaches for specific T&E events/activities rather than a single integration approach.

Whenever feasible, T&E events should be combined, if that supports technical and operational test objectives to gain the optimum amount of testing benefit for reasonable cost and time. The concept is to conduct a single, combined test program that produces credible qualitative and quantitative information that can be used to address developmental and operational issues. Similarly, capability T&E should be taken into account when further T&E schedule compression may be critical to the early fielding of a system component that provides an immediately available operational utility. Concurrent T&E should only be considered when test planners are able to identify those test events that may provide information useful to development and operational testers.

Table 2. Integrating Test & Evaluation Approaches: Advantages and Limitations

Combined Test & Evaluation	
Combined T&E refers to a single test program conducted to support objectives for multiple T&E types. Combined T&E is considered when there are time and cost savings. If selected, planning efforts must be coordinated early to ensure data are obtained to satisfy the needs of multiple agency testers and not compromise any test agency objectives.	
Advantages	**Limitations**
• Shortens time required for testing and, thus, the acquisition cycle. • Achieves cost savings by eliminating redundant activities. • Early involvement of OT&E personnel during system development increases their familiarity with system. • Early involvement of OT&E personnel permits communication of operational concerns to developer in time to allow changes in system design.	• Requires extensive early coordination. • Test objectives may be compromised. • Requires development of common test database. • Combined testing programs are often conducted in a development environment. • Test will be difficult to design to meet DT and OT requirements. • The system contractor is prohibited by law from participating in IOT&E. • Time constraints may result in less coverage than planned for OT&E objectives.
Capability Test and Evaluation	
Capability T&E is a blend of combined and concurrent T&E in that a single test program is conducted to support objectives for multiple T&E types. In capability T&E, multiple T&E types may take place at the same time as parallel and combined activities. The integration of multiple T&E types requires early planning and coordination of the test efforts to ensure that collected test data meets multiple needs.	
Advantages	**Limitations**
• Shortens time required for testing and, thus, the acquisition cycle. • Achieves cost savings by eliminating redundant activities. • Focuses on operational utility. • Early involvement of OT&E, Interoperability, and IA personnel during system development increases their familiarity with system. • Integration of multiple T&E types permits early communication of concerns to the developer that foster cost-effective changes in system design. • Provides earlier feedback to the development process.	• Requires extensive early coordination. • Test objectives may need to be negotiated between the test activities. • Requires development of common test database. • Test environment may not be fully operationally realistic. • If system design is unstable and far-reaching modifications are made, OT&E must be repeated. • Test design to meet multiple T&E type requirements is a challenge. • The system contractor is prohibited by law from participating in IOT&E. • Time constraints may result in less coverage than planned for OT&E/Interoperability/IA objectives.
Concurrent Test & Evaluation	
Concurrency is defined as an approach to system development and acquisition in which phases of the acquisition process that normally occur sequentially overlap to some extent. Concurrent T&E refers to circumstances in which multiple T&E types take place at the same time as parallel, but separate and distinct, activities.	
Advantages	**Limitations**
• Shortens the time required for testing and, thus, the acquisition cycle. • Achieves cost savings by overlapping redundant activities. • Provides earlier feedback to the development process.	• Requires extensive coordination of test assets. • If system design is unstable and far-reaching modifications are made, OT&E must be repeated. • Concurrent testing programs often do not have DT data available for OT&E planning and evaluation. • Contractor personnel frequently perform maintenance functions in a DT&E. Logistics support by user must be available earlier for IOT&E. • Limited test assets may result in less coverage than planned for OT&E objectives.

The T&E Strategy of a system acquired using evolutionary acquisition should address each increment intended for fielding. In general, T&E that has previously confirmed the effectiveness and suitability of a previous increment need not be repeated in its entirety to confirm that the subsequent increment still provides those mission capabilities previously confirmed. However, regression testing to reconfirm previously tested operational capabilities and/or suitability might be required if the subsequent increment introduces a significantly changed hardware or software configuration, or introduces new functions, components, or interfaces that could reasonably be expected to alter previously confirmed capabilities. Tailor test content and reporting against earlier test results, evaluating at a minimum the increment of mission accomplishment and survivability required of the new increment, plus whether or not performance previously demonstrated by the previous increment has been degraded.

The use of commercial items and Non-Developmental Items (NDIs) in a system acquisition can provide considerable time and cost savings and also influence the overall T&E strategy. The testing approach used must be carefully tailored to the type of system, levels of modifications, technology risk areas, and the amount of test data already available. The T&E community must get involved early in the process so that all test issues are adequately addressed and timely comprehensive evaluations are provided to decision authorities.

Test planning for commercial and NDIs shall recognize commercial testing and experience, but nonetheless determine the appropriate T&E needed to ensure effective performance in the intended operational environment. The *Defense Acquisition Guidebook* suggests that "the PM shall develop an appropriate T&E strategy for commercial items to include evaluating commercial items in a system test bed, when practical; focusing test beds on high-risk items; and testing commercial item upgrades for unanticipated side effects in areas such as security, safety, reliability, and performance." T&E must be considered throughout the acquisition of a system that involves commercial items and NDI.

The TES for programs using commercial items or NDIs will be influenced by the availability of approved T&E issues and criteria; contractor provided descriptions of the testing previously performed on the commercial items or NDIs, including test procedures followed, data, and results achieved; identification of PQT and quality conformance requirements; and acceptance test plans for the system and its components. The amount and level of testing required depends on the nature of the commercial item or NDI and its anticipated use; it should be planned to support the design and decision process. At a minimum, T&E should be conducted to verify integration and interoperability with other system elements. All commercial item and NDI modifications necessary to adapt them to the intended operational system environment are also subject to T&E. Available test results from all commercial and government sources will determine the actual extent of testing necessary. For example, a commercial item or NDI usually encompasses a mature design. The availability of this mature design contributes to the rapid development of the logistics support system that will be needed and evaluated during OT&E. Additionally, more "production" items are available early for use in an integrated T&E program.

Since an important advantage of using a commercial item or NDI acquisition strategy is reduced acquisition time, it is important that TES reduce redundant testing and that T&E is limited to the minimum effort necessary to obtain the required data. If it is determined that more information is needed after the initial data collection from the below sources, commercial items or NDI candidates may be bought or leased, and technical and operational tests may be conducted. Testing can be minimized by: (1) Obtaining and assessing contractor test results; (2) Obtaining usage/failure data from other customers; (3) Observing contractor testing; (4) Obtaining test results from independent test organizations (e.g., Underwriter's Laboratory); and (5) Verifying selected contractor test data.

All testing conducted after the program initiation milestone decision to proceed with the commercial item or NDI acquisition should be described in the TES and the TEMP. T&E continues after the system has been fielded and takes the form of an evaluation to validate and refine: operating and support cost data; Reliability, Availability, and Maintainability (RAM); logistics support plans; and training requirements, doctrine, and tactics.

T&E is the key mechanism for capturing the quantitative data used to evaluate development progress and readiness for production and deployment. Accordingly, test must be structured in a way that ensures that the necessary and sufficient conditions are satisfactorily demonstrated in a timely, relevant manner. It should also be noted that while the focus of development and operational test may be different, they have a common objective - ensuring that the user obtain systems that are reliable, supportable, and suitable for their needs. Recognizing that test is a continuous spectrum of activities across the acquisition life-cycle, test strategies may be best categorized by the basis for the criteria that will used to evaluate system performance. From this perspective, there are three test focuses: end-game, adaptive, and progressive.

Table 3. Test and Evaluation Focuses: Advantages and Limitations

Advantages	Limitations
End-Game — Concentrates on the total performance, suitability, and supportability requirements that are to be satisfied by the system. It looks at and evaluates the system from the perspective of final desired parameters.	
• Most fitted for Single Step to Full Capability with development strategy that is either waterfall or incremental development without incremental production and deployment. • Can be used successfully for evolutionary or product improvement acquisitions when: the end-game criteria are "down-scaled" to reflect the desired available performance, suitability, and supportability characteristics, not those of the "completed" system; and the demonstrated characteristics are assessed from the perspective that quantifiable evidence is present to support a claim that "completed" system characteristics are achievable in a cost- effective manner.	• Assumes the existence of fixed, well defined requirements and operational concepts. • Anything less than full compliance with operational requirements can be rated, at best, marginal. • Places an additional pressure on the acquisition management and user communities when "marginal performance" assessments can be used to satisfy political agendas. • Using an end-game focus for evolutionary or product improvement acquisitions can increase the perception of risk in the management decision-making process, even when the demonstrated core or initial capability fully meets, or exceeds, expectations at a particular point in the acquisition life-cycle.
Adaptive — Focuses on performance, suitability, and supportability requirements for a capability, not a full system, which can be provided to satisfy a mission need or a portion of that need that has operational utility.	
Advantages	**Limitations**
• Provides mechanisms for assessing a capability and identifying those mission elements that would be satisfied by early fielding of that capability. • Properly implemented, it ensures that the evaluation process directly relates available performance, suitability, and supportability levels to a defined military need or portion of that need.	• Does not necessarily keep the total system capability in mind. • Places an additional pressure on the acquisition management, T&E, and user communities to provide operational utility rather than systems. • Assumes that it is acceptable to partially satisfy requirements and operational concepts.
Progressive — Captures the desired total system requirements and characteristics and decomposes them into a series of criteria that progress from the interim to the core/initial capability to total system level.	
Advantages	**Limitations**
• Provides mechanisms for assessing development progress, identifying risk elements early, and maintaining test result and evaluation integrity. • Properly implemented, it ensures that the evaluation process directly relates available performance, suitability, and supportability levels to defined core/initial, follow-on, and total system capabilities as the system matures.	• It must keep total system capability in mind, and routinely provide demonstrated evidence of progress towards and likelihood of achieving total system capability.

Though best fitted for evolutionary or preplanned product improvement acquisitions, the progressive focus can be incorporated into any acquisition and development strategy. Development test normally utilizes a progressive test strategy. This is due to the availability of test assets at the component level early during the development process and the subsequent integration and test of upper-level components as the system matures. This approach also enables development test results to be an integral part of the system's risk mitigation strategy. The adaptive focus is just emerging and is primarily used for the rapid development and deployment of capabilities rather than systems. Operational test normally uses an end-game test strategy. Accordingly, for evolutionary and product improvement acquisitions, it is imperative that the development, operational test, and user communities define the core/initial capability, follow-on, and total system characteristics in a manner that: retains the integrity of operational test results; and presents evaluations of available capabilities in the context of the desired capabilities for a particular point in the system's life-cycle.

It can be contended that the keys to effective planning and execution are: credibility, accountability, and strategies that work together. Further, these elements must contribute to management's understanding of the development effort's health by answering four critical questions: (1) "Where are you trying to go?"; (2) "How are you going to get there?"; (3) "Where are we now?"; and (4) "How good are we?" This understanding can then be used to identify critical paths, to realistically assess development schedules, and to provide timely, relevant and meaningful information to decision makers. Further, history has shown that if any of these questions can not be answered, there is a high likelihood that the program is in trouble. The answers to the first two questions lie in credibility, while the answers to the latter two questions lie in accountability.

CREDIBILITY

Credibility is the key ingredient in implementing a streamlined approach to weapon systems acquisition. The system's developing agency must establish credibility with multiple organizations, primarily the operational user (the customer), the logistics community, independent test agency, and Service staff, Joint Chiefs of Staff, Office of the Secretary of Defense staff, the Congress, and the contractor(s). Credibility is exhibited through the existence and use of the various planning (the how you are going to get there) and requirements (the where are you trying to go) documents.

Credibility Axiom 1: If Don't Know Where You Are Going And Don't Care How You Going To Get There, Any Map Will Do.

If a developing agency possesses credibility, the development process lends itself to a streamlined acquisition. Conversely, the lack of credibility is an inherent inhibitor of acquisition streamlining. Some examples of how a lack of credibility manifests itself may prove useful in understanding how pervasive it can be:

- A system where the schedules were unrealistic (including non-achievable Initial Operational Capability (IOC) dates), requirements (developmental and operational) ill-defined, and a funding profile that does not support appropriate system development, test, production, and support capabilities to field the system;
- System proponents (government or industry), who knew what funds were available, were informed by their staff element that the available funds and/or schedule were insufficient to achieve the professed system requirements, and still promised the world that the total system could be developed and fielded within available resources;
- Aa development schedule that was driven by end dates (arbitrary or politically inspired IOCs) instead of a schedule that reflected what was really necessary to meet the requirements (assuming that they were actually understood, defined, and baselined); and
- A system that entered Engineering and Manufacturing Development with a plan of three years to deployment and then watched the development time grow to 5, 7, 11, or 15 years because requirements kept changing, or were poorly understood, defined, or baselined.

The existence of planning and requirements documents alone is not a necessary and sufficient condition for establishing credibility. It is also vital that the strategies (acquisition, development, contracting, test) work together. There is an old, but still true, saying that those who fail to learn from the past are doomed to repeat it. As systems become increasingly complex combined with the continued increase in competition for scarce national resources, the challenge is to ensure that the acquisition, contracting, development, and test & evaluation strategies work together to promote system success. The chosen strategies must focus on increased customer/user satisfaction by providing products that meet their needs. They must bring realism to the affordability, executability, and system assessments vital to acquisition decisions and promote an affordable balance of the quality and quantity of military weapons and systems.

Credibility Axiom 2: Rules of Thumb:

1. **Most lessons-learned have there roots in poorly defined requirements and management.**
2. **The trouble with lessons-learned is that nobody learns the lesson.**
3. **In case of doubt, see Rule 1.**

Strategy Integration

Effective and efficient strategies help us to learn more about a system by focusing our attention on the critical technical and operational requirements, or in other words more "bang" for the "buck." With a continuing decrease in discretionary resources and competing system solutions, efficient and cohesive strategies are a must. The strategies must be determined, executed, and analyzed for their focus on productive solutions to acquisition concerns and/or issues, particularly important when dealing with evolutionary acquisitions. The demonstration of interim performance thresholds is a key factor in establishing a path to system success.

The chosen strategies must mature as the program proceeds through the acquisition process to ensure acquisition and fielding of an effective and supportable system by providing information to management, feedback to the engineering design and development process, and verifying attainment of technical performance specifications and supportability, suitability and operational effectiveness objectives. Feedback is essential not only to the development community, but also is a key contributor in the improvement of operational doctrine and tactics. This is particularly evident in the "high tech" arena where technology solutions are procured before user applications have solidified.

Acquisition, contracting, development, and T&E strategies must be determined in a way that provides credible, viable, value-added information to support the decision-making process. The strategies must focus on critical processes that affect customer satisfaction, mechanisms for characterizing the process, tracking progress, provide feedback, and respond to management. It is imperative that the strategies be in concert with each other. Inconsistencies do not promote either management or user confidence in the likelihood of program success. Table 1 provides an overview of how elements of the various strategies work together (or don't work together. Some examples of inconsistent strategies include the following:

A. Evolutionary Acquisition - Waterfall: This combination is inherently incompatible, particularly when looking at the basic assumptions concerning requirements. Waterfall (or grand design) assumes fixed, well defined requirements; while evolutionary acquisition assumes that requirements can be broadly defined and need further refinement over time. While a modified waterfall strategy may be appropriate for a particular core capability within the acquisition life-cycle, using it for evolutionary acquisition normally leads to program failure.

B. Product Improvement - Waterfall: Though not as incompatible as above, this combination is not well suited for technology risk management. Waterfall typically assumes mature, existing technology; while preplanned product improvement recognizes up-front that all system requirements may not be satisfied initially and that provisions must be designed into the system to promote later incorporation of deferred capabilities as technology matures. A modified grand design approach may be appropriate for the initial and deferred capabilities, but its unmodified use does not encourage successful program acquisition and development.

C. Single Step to Full Capability - Incremental Development With Incremental Production and Deployment: This is an incompatible combination Single Step to Full Capability acquisition presumes fixed, well defined requirements, mature technology, and a single deployable configuration, while the incremental development, production, and deployment assumes just the opposite. It should be noted, however, that some Single Step to Full Capability acquisitions may transition to this development strategy in order to control risks that were not foreseen at the beginning of the acquisition process.

D. Evolutionary Acquisition - Incremental Development Without Incremental Production and Deployment: Incremental development is compatible with evolutionary acquisition, however, the single production and deployment aspects make this combination incompatible. It also tends to delay getting new operational capabilities into the hands of the user to promote the feedback mechanisms for requirements and operational concepts refinement that are an integral part of evolutionary acquisition. Control of technology risk in the transition to production, i.e., trying to prevent the forced integration of immature technology into a production configuration, can give management and the user the impression that the core capability is unachievable.

E. Preplanned Product Improvement - Incremental Development Without Incremental Production and Deployment: As above, the single production and deployment aspects make this combination incompatible.

Table 1. Strategies that Work Together

Legend:

👍 - Positive Relationship (Works well together)

☞ - Neutral/"It Depends" Relationship

👎 - Negative Relationship (Does not work well together)

		Life-Cycle Models				Contract Types							Acquisition Strategies			Test & Evaluation Strategies		
		Waterfall	Incremental	Phased/Evolutionary	Spiral/Iterative	FFP	FPI	FPAF	CPFF	CPIF	CPAF	T&M	Evolutionary Acquisition	Product Improvement	Single Step to Full Capability	Progressive	Adoptive	End-Game
Life-Cycle Models	Waterfall					👍	👍	👍	👎	👎	👎	☞	👎	☞	👍	👎	👎	👍
	Incremental					☞	☞	☞	☞	☞	☞	☞	☞	☞	👎	👍	☞	👎
	Phased/ Evolutionary					👎	👎	👎	👍	👍	👍	☞	👍	☞	👎	👍	👍	👎
	Spiral/Iterative/Agile					👎	👎	👎	👍	👍	👍	☞	👍	☞	👎	👍	👍	👎
Contract Types	FFP	👍	☞	👎	👎								👎	☞	👍	👎	👎	👍
	FPI	👍	☞	👎	👎								👎	☞	👍	👎	👎	👍
	FPAF	👍	☞	👎	👎								👎	☞	👍	👎	👎	👍
	CPFF	👎	☞	👍	👍								👍	☞	👎	👍	👍	☞
	CPIF	👎	☞	👍	👍								👍	☞	👎	👍	👍	☞
	CPAF	👎	☞	👍	👍								👍	☞	👎	👍	👍	☞
	T&M	☞	☞	☞	☞								☞	☞	☞	☞	☞	☞
Acquisition Strategies	Evolutionary Acquisition	👎	☞	👍	👍	👎	👎	👎	👍	👍	👍	☞				👍	👍	👎
	Product Improvement	☞	☞	☞	☞	☞	☞	☞	☞	☞	☞	☞				☞	☞	☞
	Single Step to Full Capability	👍	👎	👎	👎	👍	👍	👍	👎	👎	👎	☞				👎	👎	👍
Test & Evaluation Strategies	Progressive	👎	👍	👍	👍	👎	👎	👎	👍	👍	👍	☞	👍	☞	👎			
	Adoptive	👎	☞	👍	👍	👎	👎	👎	👍	👍	👍	☞	👍	☞	👎			
	End-Game	👍	👎	👎	👎	👍	👍	👍	☞	☞	☞	☞	👎	☞	👍			

From a T&E perspective, there are some additional questions that can be asked to ensure a program's credibility. A typical list of questions that could be asked for each test phase includes:

- What organization has the responsibility for test and test result evaluation?
- What testing activities are included in the T&E plans and schedules?
- Are the goals of the activities that will be used to measure the degree of success of the T&E effort defined?
- Are the inputs to the activities identified?
- What entrance criteria or qualifiers are included in the inputs to the T&E activities?
- If the entrance criteria are not satisfied, should the testing effort be postponed because there is a risk that the testing effort will need to be repeated?
- Are the outputs from the T&E activities defined?

- What exit criteria or quantifiers are on the outputs of the testing activities?
- For any given test phase, is the T&E sufficient to verify that the exit criteria are satisfied?

ACCOUNTABILITY

The answers to the "Where are we now?" and "How good are we?" questions lie in the ability to realistically determine program status (the where) and integrate the results of multiple assessment activities (the how good). This requires that, at a minimum, the following assessment activities/tools be considered/used:

1. Earned value measurement – confirm what we already know;
2. Quality Assurance – verify that the defined processes are being followed;
3. Quality Control – validate that the defined processes are producing products that meet expectations; and
4. Test & Evaluation – assure that the products are meeting specifications and providing quantitative evidence of maturity towards satisfying user requirements.

The implementation of accountability is a difficult process, with multiple levels. So let's look at several ways that show that accountability is not in place.

Accountability Axiom 1. Program Advocates Do Not Kill Their Own Systems, Or No News Travels Faster Than Bad News.

A good example of a lack of accountability is the inability of some programs to adopt the concept of a "successful" failure, i.e., admit something went wrong and learn from it (fix the problem). Examples of this include: not calling a problem what it is, call it a "management opportunity;" don't disapprove something; return it for "clarification;" and always accentuate the positive. The key issue is that programs are not allowed to fail during development. Failures are reclassified, or the success criteria changed, so that no one has to admit that they are having problems to upper and senior management.

Accountability Axiom 2. The Problem With Lessons Learned Programs Is That We Haven't Learned The Lessons.

One way to promote information flow and instill credibility into the acquisition process is to recognize that there are two different types of failure: Successful Failures and Unsuccessful Failures. Successful failures occur everyday in our lives. We all make mistakes, but if we learn from them, change our strategies to incorporate what we have learned, and progress, we have been successful. Successful failures should not be hidden; they should be shared so that others do repeat them.

Accountability Axiom 3. The Threat and Political Environment Change Faster Than Our Ability To React.

New systems are all started due to changes in the threat or political environment. Projections are made of when the systems are needed to counter the full threat or change the political environment and an IOC established. Does a system have to work at IOC to counter the full threat? No! So long as the new system is better that what we have, is operationally reliable, improves the ability of the service members in the field to stay alive and continue the fight, and growth capability to meet the full or changed threat.

Accountability Axiom 4. The Developer Is Accountable To The User And Not Vice Versa.

This leads to one member of the acquisition structure who is not accountable, in the acquisition sense, to anyone - the User. Users always want everything yesterday, systems that counter the full threat, and changes instantaneously incorporated to reflect changes in the threat. They have to be prepared to fight the war and want to have the capabilities to win. This is perfectly logical on the user's part. However, their needs must be balanced with development realities and they must recognize that continuous requirement changes do not promote getting better, more reliable, and operationally responsive systems faster.

Acquisition Oversight

A quantitative assessment structure is driven by the information needed to address management issues. Specific information needs and concerns determine the selection and application of attribute measures and the analysis focus. These can and will change over the program's life cycle. The overall measurement process is intended to support the immediate and projected information needs of the program manager. Information needs are categorized as program, process, product, and risk. The data, indicators, and analysis results support the questions defined below and provide for objective communications between the acquirer and the developer.

Program Oriented Questions. Program planning, execution, and status are of key interest to the program manager. Program plans, e.g., Systems Engineering Management Plan, Software Development Plan, Continuous Process Improvement Plan, etc., establish process whose purpose is to satisfying the technical and operational performance objectives of the development effort. Execution translates the plan into actual processes being used and refines the measures of merit that will be used to validate the achievement of performance (quality) and technical objectives. Status provides the program manager with the current status in the development process and related product quality, and the ability to predict when the program will be completed based upon quantitative evidence. Sample questions include:

Question	Basis of Answer
What is the status of the program?	This question will be answered in terms of work breakdown structure (WBS), schedule, and the resource plan/action.
For what purpose is the program intended?	The answer will address requirements traceability, requirements understanding, and end product validation.
What are the customer's needs?	This will consider the urgency associated with satisfying the customer's needs and the identification of the appropriate product quality factors allied with the requirements.
How stable are the user's requirements?	The question will be answered using the results of the requirements analysis process, including the determination of stated and derived requirements, affects the translation of these requirements into design and product specifications.
Is there early identification of user requirements and changes?	Related to the requirements analysis process, this question focuses on customer needs, both stated and derived, and the change control process that can affect program development and customer satisfaction.
Is the product mature enough to be used in operation?	Maturity will be addressed in terms of product quality metrics, readiness for test and evaluation, and readiness for fielding and/or deployment.

Process Oriented Questions. The process defines the set of activities that will be performed to produce the desired products. The program manager is typically interested in the resources associated with the process (including cost, schedule, and resources) and how the process is impacted by scope and requirements changes. Sample process questions include:

Question	Basis of Answer
Are costs under control?	Funds availability, estimated costs to completion, and tracking costs against planned expenditure rates form the key elements of cost control.
Is the project on schedule?	Schedule aspects may be tracked at the upper level and specific activity level, e.g., design, code, test.
Is the project within resource limitations?	Resource limitations can apply to bandwidth, throughput, input/output constraints, etc.
Are scope changes impacting the project?	Potential scope changes include any change to the deliverable products and tasks contained in the SOW.
Are requirement changes impacting the project?	Requirement changes include those presented by the user and those derived during the requirements analysis and allocation process.
Is the installation and/or user ready to receive the product?	Have the necessary physical, workflow, and organizational modifications been identified and made to use the product?

Product Oriented Questions. All development and sustaining engineering activities produce products. Products include, but are not limited to, designs, documentation, executable code, prototypes, software systems, source code, and specifications. The program manager is interested in the quality of these products. Sample product questions include:

Question	Basis of Answer
Are defects being identified and corrected?	Defect tracking can provide insight into the source of defects, defect discovery rates, and defect closure rates.
Is the product of high quality and is the end user satisfied?	With quality being defined in terms of one or more of the following quality attributes: maintainability, correctness, reusability, testability, portability, reliability, adaptability, availability, transportability, flexibility, and/or safety.

Risk Oriented Questions. Identifying, assessing, and managing risk is a key program management function. As visibility increases into the development process and the resultant product quality, program management can make maximum use of collected data to support system risk management. To support this function, typical risk related questions include:

Question	Basis of Answer
What is the technical risk?	Technical risk is the cornerstone of risk control and affects the inherent risks for all other risk areas. Technical risk control includes determination of the performance envelope of the development and operational software within the context of the overall system.
What is the cost risk?	Cost estimates, normally deterministic, are fundamental to program management. Costs for individual program elements are usually expressed as a single value representing the "best" available estimate.
What is the schedule risk?	Schedule risk control includes determination of the range of times covering the span between earliest and latest dates for program completion, with detailed schedules and risks
What is the operational risk?	Using a range of estimates, identifying the most likely outcome, and assessing the risk of having a system that does not meet requirements in the customer's operational environment are all components of operational risk control.
What is the support risk?	There are several issues which complicate risk control in the sustaining engineering (support) environment. Among these issues are: determination of what facilities, personnel, and procedures are necessary for sustaining engineering; and what re-sources are available for sustaining engineering.

Metrics for Answering the Questions

The measurement process should be flexible and tailorable to distinct development program characteristics, objectives, and limitations. It should be adapted to different development process models and implementations. Once the measurement process is tailored to meet the specific information needs of an individual program, it should be consistently applied throughout the development or sustaining engineering process.

Attribute measures are not constrictive. Within the overall measurement process, different measures can be applied for different programs. Attribute measurement methodologies should be well defined and consistently applied across the development. The actual measures applied during the development evolve with respect to changing development process activities, development products, and information requirements. Measurement should be implemented at the lowest practical level. Multiple measures can be applied with to a given issue to support cross-validation of the findings.

Generic information categories (process characteristics, product characteristics, product quality, progress, and resources) that can be used to answer the above questions are presented in Table 1. Since these information categories are multi-dimensional, metric categories that can be used to address the specific attributes of each information category are also identified. These information categories can, in turn be related to the metric categories and individual metrics that can supply the necessary data that through the data analysis process provide the necessary information to meet the decisionmaker's information needs.

INFORMATION CATEGORIES	Process Characteristics:	Product Characteristics:	Product Quality:	Progress:	Resources:
ATTRIBUTES	Applicable Metric Categories				
Complexity		Size & Complexity			
Configuration	Stability	Stability	Stability		
Cost	Management			Management	Management
Defects		Testing	Testing		
Earned Value		Stability		Stability	
Environment	Management			Management	Management
Productivity	Management		Testing	Management	Management
Progress	Stability	Stability	Testing	Stability	
Quality		Features	Features		
Requirements	Stability	Stability	Stability		
Schedule	Management			Management	Management
Size	Size & Complexity	Size & Complexity	Size & Complexity	Size & Complexity	
Stability	Stability	Stability	Stability	Stability	Stability
Staffing	Management		Management	Management	Management
Test	Testing	Testing	Testing	Testing	

Table 1. Multi-Dimensional Information Requirements Are Satisfied by Metrics.

Just as the information categories are multi-dimensional, so is the relationship of the information categories to the various project, process, product, and risk questions. This relationship is shown in Table 2. The multi-dimensionality of the information categories and the answers to the questions provides the different perspectives necessary to prevent misinterpretation of a single metric or metric category. This built in cross-checking also promotes a return on investment of the metrics data collection costs by making maximum use of the collected data. It also provides a structure that present the information in terms that are useful in the management, e.g., oversight, program, engineering, etc., decisionmaking process and is not too detailed or too sketchy so that it doesn't lose its value to the decisionmaker.

Measurement Goal: To provide decisionmakers with the necessary information that supports the project's productivity, quality, process improvement, and risk management objectives.					
Questions	**Information Requirements**				
	Process Characteristics	**Product Characteristics**	**Product Quality**	**Progress**	**Resources**
Project Questions					
1. What is the status of my program?	X	X	X	X	X
2. For what purpose is the program intended?		X			
3. What are the customer's needs?		X			
4. How stable are the user's requirements?		X			
5. What are the risks to my program?	X	X	X	X	X
6. Is there early identification of user requirements and changes?		X			
7. Is the product mature enough to be used?		X	X	X	
8. When do I freeze my requirements and/or design?	X	X	X	X	X
Process Questions					
1. Are costs under control?	X			X	X
2. Is the project on schedule?	X	X	X	X	X
3. Is the project within resource limitations?		X			X
4. Are scope changes impacting the project?		X		X	X
5. Are requirement changes impacting the project?		X		X	X
Product Questions					
1. Are defects being identified and corrected?		X	X	X	
2. Is the product of high quality?		X	X	X	
Risk Questions					
1. What is the technical risk?		X	X		
2. What is the cost risk?	X			X	X
3. What is the schedule risk?	X			X	X
4. What is the operational risk?		X	X		
5. What is the support risk?		X	X		X

Table 2. Information Categories Related to the Program Manager's Questions

Objective evaluation of development processes and products uses a quantitative development assessment structure. The assessment process is based upon quantitative measures (metrics) of process and product attributes applied across the life cycle. These attributes are evaluated within the assessment structure with respect to program characteristics, constraints, and objectives, and integrated into an overall assessment of the process and resultant products within the overall context of the program.

The program manager's information needs are satisfied by multiple disciplines. The metrics used provide the information that can be used by these disciplines in a form that is compatible with the normal information analysis methods and tools used by these disciplines (Table 3). Use of product information by these other disciplines fosters the awareness of product impacts on these disciplines and promotes the presentation of the product information within a system context.

Figure 1 identifies specific activities that can be conducted during any phase of the life cycle. These activities can be grouped into six top level activities: requirements analysis, design, implementation, integration and test, support and delivery, and integral and common activities. The generic metric types can be directly related to the top level activities, with specific data primitives that can be collected during each activity. Data primitives can then be input into specific metrics that are chosen by the program office to meet management's information needs. The chosen metrics can be used to provide the feedback and status information necessary to continuously monitor and improve the process and products.

DISCIPLINES	ACTIVITIES	APPLICABLE METRICS
POLICY AND ORGANIZATIONAL MANAGEMENT		
ACQUISITION POLICY	ACQUISITION PHASES	MANAGEMENT STABILITY SIZE and COMPLEXITY TESTING
	ACQUISITION STRATEGY	
	DECISION POINTS	
PRINCIPLES OF PROGRAM MANAGEMENT	PLANNING	MANAGEMENT STABILITY
	ORGANIZING AND STAFFING	
	CONTROLLING AND EXECUTING	
MANAGERIAL DEVELOPMENT	INTERPERSONAL COMMUNICATION	MANAGEMENT
	PROGRAM MANAGEMENT TEAM	
	PERSONNEL MANAGEMENT	
BUSINESS MANAGEMENT		
CONTRACTOR FINANCE	FINANCIAL ANALYSIS AND FORECASTING	MANAGEMENT STABILITY
	FINANCIAL REPORTING AND COST MANAGEMENT	
	CONTRACT PRICING	
COST/SCHEDULE MANAGEMENT	CONTRACT FUNDS STATUS REPORTING	
	COST PERFORMANCE REPORTING	
	PROGRAM LIFE CYCLE COST ESTIMATE	
	WORK BREAKDOWN STRUCTURE	
CONTRACT MANAGEMENT	CONTRACT TYPE SELECTION	
	RFP PREPARATION	
	SOURCE SELECTION	
	CONTRACT EXECUTION	
TECHNICAL MANAGEMENT		
SYSTEMS ENGINEERING	SYSTEM REQUIREMENTS ANALYSIS	MANAGEMENT FEATURES STABILITY SIZE AND COMPLEXITY TESTING
	CONFIGURATION MANAGEMENT	
	DATA MANAGEMENT	
	SYSTEM DESIGN	
	SYSTEM INTEGRATION	
	SYSTEM PERFORMANCE ASSESSMENT	
SOFTWARE MANAGEMENT	SOFTWARE REQUIREMENTS ANALYSIS	
	SOFTWARE DESIGN	
	SOFTWARE INTEGRATION	
	COMPUTER RESOURCES	
	SOFTWARE PERFORMANCE ASSESSMENT	
TEST AND EVALUATION	DEVELOPMENT TEST	
	OPERATIONAL TEST	
LOGISTICS SUPPORT	LOGISTICS SUPPORT ANALYSIS	
	SUPPORT CONSTRAINTS	
	POST PRODUCTION SUPPORT	

Table 3. Metrics Can Be Used By Multiple Disciplines.

TOP LEVEL ACTIVITIES	**REQUIREMENTS ANALYSIS**	**DESIGN**	**IMPLEMENTATION**	**INTEGRATION & TEST**	**DELIVERY and SUPPORT**
Activities	System requirements analysis Hardware/Software requirements analysis	System design Hardware design Software design	Hardware/Software implementation and testing	Unit integration and testing CI Qualification testing CI Integration and testing System qualification testing	Preparation for use Preparation for transition
Integral & Common Activities	Project planning Development environment Product evaluations Configuration management Joint technical and management reviews Corrective action Quality assurance Other management activities				
INFORMATION CATEGORIES	**APPLICABLE METRIC CATEGORIES**				
Process	Management		Size and Complexity	Stability	Testing
Product	Management	Features	Size and Complexity	Stability	Testing
Product Quality	Management	Features	Size and Complexity	Stability	Testing
Progress	Management		Size and Complexity	Stability	Testing
Resources	Management	Features	Size and Complexity	Stability	

Figure 1. Top Level Activities and Applicable Metric Categories

The primary goal of an **Acquisition Strategy** is to minimize the time it takes to satisfy an identified need consistent with common sense, sound business practices, and the provisions of DoDD 5000.01 and DoDI 5000.02. Both documents recognize the complementary evolution during the life-cycle of requirements definition, affordability constraints, and weapon system design. During the acquisition life-cycle, broad objectives and minimum acceptable requirements, affordability goals, and alternative concepts are established at Milestone A and progressively refined to more detailed and specific objectives, requirements, and thresholds, unit costs, and stable design at successive milestone decision points. The intent of this life-cycle process is to keep reasonable options open, facilitate cost-schedule-performance trade-offs early in the process, and avoid the premature commitment to a system-specific solution.

The acquisition strategy also matures through an iterative process and becomes more definitive as the program proceeds through the acquisition process. Alternative acquisition strategies are to be considered when requirements refinements are anticipated or when a technology risk/opportunity discourages implementation of a required capability. It should be noted, that neither DoDD 5000.01 nor DoDI 5000.02 specifically addresses the types of development or test strategies associated with the acquisition strategy.

The acquisition strategy results from extensive planning and preparation and a thorough understanding of both the specific acquisition program and the general defense acquisition environment. Development of the acquisition strategy requires collaboration between the Milestone Decision Authority, program manager, and the functional communities engaged in and supporting DoD acquisition. A well-developed strategy minimizes the time and cost required to satisfy approved capability needs, and maximizes affordability throughout the program life-cycle. Consistent with DoDD 5000.01, the program manager is the single point of accountability for accomplishing program objectives for total life-cycle systems management, including sustainment. The charge of DoD executive leadership is to use common sense and sound business practice in developing the acquisition strategy and executing the program.

Key components of an Acquisition Strategy for a DoD program include the following items:

- **Acquisition Approach** defines either the evolutionary or single step approach the program will use to achieve full capability; it should include a brief rationale to justify the choice. The DoD preference is evolutionary acquisition. When a program uses an evolutionary acquisition strategy, each increment should have a specific set of parameters with thresholds and objectives appropriate to the increment.
- **Best Practices** address management constraints imposed on contractors. This section should identify the best practices, for example, Integrated Product and Process Development; performance-based specifications; management goals; reporting and incentives; a modular open systems approach that emphasizes modularity and use of commercially supported practices, products, performance specifications, and performance-based standards; replacement of Government-unique management and manufacturing systems with common, facility-wide systems; technology insertion for continuous affordability improvement throughout the product life cycle; realistic cost estimates and cost objectives; adequate competition among viable offerors; best value evaluation and award criteria; the use of past performance in source selection; results of software capability evaluations; Government-Industry partnerships, that implemented and the use of pilot programs to explore innovative practices.
- **Capability Needs Summary** presents a top-level description of the capability the acquisition is intended to satisfy or provide. The summary should highlight system characteristics driven by interoperability and/or joint integrated architectures, capability areas, and families or systems of systems. The summary should also identify any dependency on the planned or existing capability of other programs or systems. The summary should state whether the approved capability need is structured to achieve full capability in time-phased increments or in a single step. For time-phased capabilities, define the initial increment, as well as subsequent increments.
- **Environment, Safety, and Occupational Health** (ESOH) Issues section provides a summary of the Programmatic ESOH Evaluation (PESHE), including a strategy for integrating ESOH considerations into the systems engineering process; ESOH risks and risk mitigation efforts; and a compliance schedule for National Environmental Policy Act (NEPA).

- **Human Systems Integration** (HSI) describes the incorporation of manpower, personnel, training, human factors, safety and occupational health, personnel survivability, and habitability considerations into the acquisition process. This portion should identify HSI responsibilities, describe the technical and management approach for meeting HSI requirements, briefly summarize the planning for each of the above elements of HSI, and summarize major elements of the associated training system.
- **Information Assurance** (IA) identifies the technical, schedule, cost, and funding issues associated with implementing information assurance. The planning for and documentation of the Acquisition IA Strategy should produce the information required for this section. Potential IA considerations to be included in the Acquisition Strategy include: overview, technical considerations, schedule considerations, cost considerations, funding considerations, and IA staffing and support issues.
- **Information Technology** summarizes the infrastructure and support considerations identified in the appropriate capability document and described in the Information Support Plan (ISP). This portion should identify Information Technology, including National Security Systems, infrastructure enhancements required to support program execution. It should identify technical, schedule, and funding critical path issues for both the acquisition program and the Information Technology, including National Security Systems, infrastructure that could affect execution of the acquisition strategy. Shortfalls and issues should be described, and plans to resolve them discussed. The Acquisition Strategy need not repeat the details found in the ISP, but should be consistent with the ISP.
- **Integrated Test and Evaluation** (T&E) summarizes T&E Strategy (TES) for the Milestone A decision and the Test and Evaluation Master Plan (TEMP) for the Milestone b and C decisions. It also presents how the TES /TEMP was harmonized with the acquisition strategy. The Acquisition Strategy need not repeat the details found in the TES/TEMP, but should be consistent with the TES/TEMP
- **Interoperability** describes the treatment of interoperability requirements. For example, if an evolutionary acquisition strategy involves successive increments satisfying time-phased capability needs, the program manager should address each increment and the transitions from increment to increment. This section should identify any waivers or deviations that have been requested, obtained, or expected to be requested. The Strategy should reflect full compliance with the interoperability considerations and, for Information Technology, including National Security Systems considerations.
- **Modular Open Systems Approach** (MOSA) provides summary of the planning that describes (1) how MOSA fits into a program's overall acquisition process and strategies for acquisition, technology development, and T&E; (2) what steps a program will take to analyze, develop, and implement a system or a system-of-systems architecture based on MOSA principles; and (3) how such program intends to monitor and assess its MOSA implementation progress and ensure system openness.
- **Product Support** describes the supportability planning, analyses, and trade-offs used to determine the optimum support concept for a materiel system and identify the strategies for continuous affordability improvements throughout the product life cycle. The support strategy evolves in detail, so that by Milestone C, it defines how the program will address the support and fielding requirements necessary to meet readiness and performance objectives, lower total ownership cost, reduce risks, and avoid harm to the environment and human health. The support strategy should address how the program manager and other responsible organizations will maintain oversight of the fielded system.
- **Program Structure** establishes the milestone decision points and acquisition phases, including development, testing, production, and lifecycle support, planned for the program. It should prescribe the accomplishments for each phase, and identify the critical events affecting program management. This section should include a summary of the Integrated Master Plan and Integrated Master Schedule.
- **Relief, Exemption & Waiver** identifies mandatory acquisition process requirements that fail to add value, are not essential, or are not cost effective, and seek the appropriate relief, exemption, or waiver.
- **Research & Technology Protection** identifies the technical, schedule, cost, and funding issues associated with protecting critical program information and technologies, and the plans to resolve them.
- **Resource Management** addresses the estimated program cost and the planned program funding, including funding under an evolutionary acquisition strategy and advance procurement.
- **Risk Management** summarizes the risk management process established by the program manager. Effective risk management depends on the knowledge gleaned from all aspects of the program.
- **Systems Engineering** summarizes the program's systems engineering strategy early in the program definition stages and is updated periodically to summarize the Systems Engineering Plan (SEP) as a

program matures. The SEP describes a program's overall technical approach, including processes, resources, and metrics, and applicable performance incentives. It should describe the systems engineering processes to be applied, the approach to be used to manage the system technical baseline, and how systems engineering will be integrated across the integrated product team structure. It should also detail the timing, conduct, entrance criteria, and success/exit criteria of technical reviews. The Acquisition Strategy need not repeat the SEP details, but should be consistent with the SEP.

The Acquisition Strategy should also briefly address the program manager's consideration of, decisions on, and planning for the following additional topics:

- ***Program Office Staffing and Support Contractor Resources Available to the Program Manager*.** The program manager should identify resource limitations that prevent the program manager from pursuing a beneficial acquisition strategy or contracting approach (e.g., component breakout (i.e., the Government contracts for a component and furnishes it to the prime contractor), or the use of an award fee contract). The program manager should provide an estimate of the additional resources needed to implement the desirable strategy or approach.
- ***Integrated Digital Environment Management*.** The program manager should summarize plans to establish a cost-effective data management system and digital environment.
- ***Government Property in the Possession of Contractors (GPPC) Management.*** *The program manager should* summarize the planned management of GPPC.
- ***Simulation Based Acquisition and Modeling and Simulation*.** The program manager should summarize the planned implementation of Simulation Based Acquisition and Modeling and Simulation during engineering, manufacturing, and design trade studies; and during developmental, operational, and live fire testing.
- ***Software-Intensive Programs Review*.** The program manager should describe the planned use of independent expert reviews for all Acquisition Category I through Acquisition Category III software-intensive programs.

When a material solution is required, DoDD 5000.01 states that the DoD Component(s) shall:

- Consider multiple concepts and analyze possible alternative ways to satisfy the user need;
- Seek the most cost-effective solution over the system's life cycle;
- Conduct market research and analysis to determine the availability, suitability, operational supportability, interoperability, safety, and ease of integration of the considered and selected procurement solutions; and
- Work with users to define capability needs that facilitate the following, listed in descending order of preference:
  1. The procurement or modification of commercially available products, services, and technologies, from domestic or international sources, or the development of dual-use technologies;
  2. The additional production or modification of previously-developed U.S. and/or Allied military systems or equipment;
  3. A cooperative development program with one or more Allied nations;
  4. A new, joint, DoD Component or Government Agency development program; or
  5. A new DoD Component-unique development program.

The Acquisition Strategy defines the management approach that will achieve material solution program goals. DoDD 5000.01 states that evolutionary acquisition strategies are the preferred approach to satisfying operational needs. *Incremental* development is the preferred process for executing such strategies. The approved Acquisition Strategy should address the proposed management approach to be used to define both the capability and the strategy applicable to each increment. This discussion should specifically address whether end items delivered under early increments will be retrofitted with later increment improvements.

There are two acquisition approaches identified in DoDI 5000.02 and a third that can be inferred as a bridge between the two identified approaches:

Evolutionary Acquisition. Evolutionary acquisition is the preferred DoD strategy for rapid acquisition of operational capability for the user. In an evolutionary approach, a need is met over time by delivering operational capability in several increments, recognizing up front the need for future capability improvements. Each increment is a militarily useful and supportable operational capability that can be developed, produced, deployed, and sustained. The objectives are to: balance needs and available capability with resources, to rapidly field the first increment, and put the capability into the hands of the user quickly. Evolutionary acquisition requires user, tester, and developer collaboration.

This is an approach where a core capability is fielded, and the system has a modular structure and provisions for future upgrades and changes as requirements are refined or become technically achievable. It is considered well suited to high technology and software intensive programs where requirements beyond the core capability can generally, but not specifically be defined. The success of the strategy depends on phased definition of capability needs and system requirements, and the maturation of technologies that lead to disciplined development and production of systems that provide increasing capability over time.

Product Improvement. This approach, while not identified in DoDI 5000.02, can be inferred as a link between the evolutionary and single step approaches. This is a phased approach for modifying existing systems with mature technology that incrementally satisfies new operational requirements in order to address the cost, risk, or relative time urgency of different elements of the system being developed. Each increment provides a significant increase in operational capability and will have its own set of threshold and objective values set by the user. With this approach selected capabilities are structured so that the system can be affordably fielded and the structured elements can be developed in a parallel or subsequent effort. It also dictates a system design with provisions, interfaces, interoperability, and accessibility integrated into the design so that the structured elements can be incorporated in a cost-effective manner when they become available.

Single Step to Full Capability. A third acquisition strategy that is identified, but not defined in DoDI 5000.02, is the single step to full capability. This approach implies firm requirements, known and/or proven technology, and the desire to satisfy user requirements in a single fielding of the deployable system. The fielded system does not necessarily have a system design with provisions, interfaces, and accessibility integrated into the design so that additional capabilities can be incorporated in a cost-effective manner at a later time.

The choice of acquisition strategy is not a decision cast in stone. As a system progresses through its life-cycle, several factors can encourage a change in acquisition strategy. For example, consider the following program evolution. At Milestone A, when user requirements, operational concepts, and technological solutions are still fluid, evolutionary acquisition may be most appropriate. At Milestone B, if requirements have stabilized but technology is not sufficiently mature to meet all the requirements, a preplanned product improvement strategy may be appropriate. Subsequent to Milestone C, the program may then consider a single step to full capability approach for upgrades. Whatever the acquisition strategy, it must be consistent with the current definition and stability of requirements, operational concepts, technology, and fiscal constraints.

Life-Cycle Models

For a particular project, an appropriate development model defines an overall "ordering" of the phases and activities and implements a engineering process. The following are four approaches for implementing the engineering process. This is not an exhaustive list of all development models but is provided to illustrate various approaches.

WATERFALL MODEL

The Waterfall model (Figure 1) is the most commonly used and known software process model. In the Waterfall model, phases of development occur serially with precise boundaries. Each phase occurs once, although iteration of part of a phase can occur for error correction.

This development model is totally end-game oriented in that it presumes that full capability can be developed and achieved in a single deployable configuration. The prerequisites for this strategy include: fixed requirements, defined operational concepts, resolute user commitment, and mature technology. The criteria used to determine development progress and readiness for production and/or deployment for grand design systems are also end-game oriented. Consequently, the program is continually assessed against full operational capabilities. Performance, cost, and schedule shortfalls, even during the early phases of the acquisition life-cycle, can lead to erosion of management and user confidence in the likelihood of program success. Variations of this approach could be used for the core capability in evolutionary acquisitions or initial capability in preplanned product improvements. The Waterfall model is applicable for low risk programs.

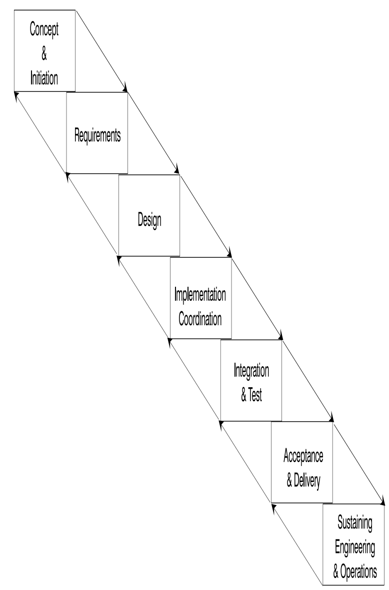
<table>
<tr><td>DESCRIPTION:
Phases of system development occur serially with precise boundaries. Each phase occurs only once, although iteration of part of the phase can occur for error correction. The evolving product is baselined at the end of each phase through formal reviews and baselining. System goes operational as a unit.</td><td rowspan="4">

Figure 1. The Waterfall Model.</td></tr>
<tr><td>SELECTION/REJECTION RATIONALE:
• Inability to incorporate additional capabilities and/or changes into the design in a cost effective manner.
• Not responsive to technology infusion
• User involvement delayed until formal delivery</td></tr>
<tr><td>STRENGTHS:
• Geared to requirements allocation for large, monolith applications
• Permits earliest imposition of baselines and configuration control</td></tr>
<tr><td>WEAKNESSES:
• Delays implementation
• Delays deployment until total system built
• Too much production of little-used documentation
• Relies on a big-bang approach to testing
• Easy to misjudge progress - 90% complete syndrome</td></tr>
</table>

INCREMENTAL MODEL

The Incremental model (Figure 2) illustrates the concept of "build a little, test a little" so that development and iteration are achieved in increments rather than all at once. The overall architectural design is determined initially, and then divided into incremental parts for partial deliveries. Again, each phase occurs once, although iteration of part of a phase can occur for error correction.

This approach allows for design changes during development to account for modifications in requirements, operational concepts, and technology maturity. It fosters management and user confidence by being responsive to change, while retaining a long term focus on customer satisfaction. As with Grand Design, the criteria used to determine development progress and readiness for production and deployment are end-game oriented due to a single production configuration. This development approach is only consistent with an unconditional acquisition strategy. Again, variations of this approach could be used for the core capability in evolutionary acquisitions or initial capability in preplanned product improvements. This model is primarily applicable for low to medium risk programs.

DESCRIPTION:
Development is achieved in increments rather than at once. Overall architecture is determined initially and divided in pieces for partial deliveries. Integration and test is achieved in stages. Delivery and acceptance of system for deployment is done after all increments are built and integrated. System goes operational as a unit.

SELECTION/REJECTION RATIONALE:
- Delays providing operational capabilities to the users
- No provisions for periodic updates of existing functionality and delivery of new capabilities to the user based on user feedback
- Lacks an overall architecture, to include process for change, which allows the system to be designed and implemented incrementally
- Does not facilitate control of program costs through separate funding approval for each increment of operational capability

STRENGTHS:
- Early implementation of parts
- Progressive baselining of pieces
- Testing is done in chunks
- Allows for design changes to account for modifications in requirements, ops concepts, and technology maturity

WEAKNESSES:
- Fragments design analysis
- Hard to review system
- Complex configuration management
- Testing is highly dependent on increments
- Test stubbing difficult
- Delays deployment until total system built
- Operational testing delayed until final increment is complete

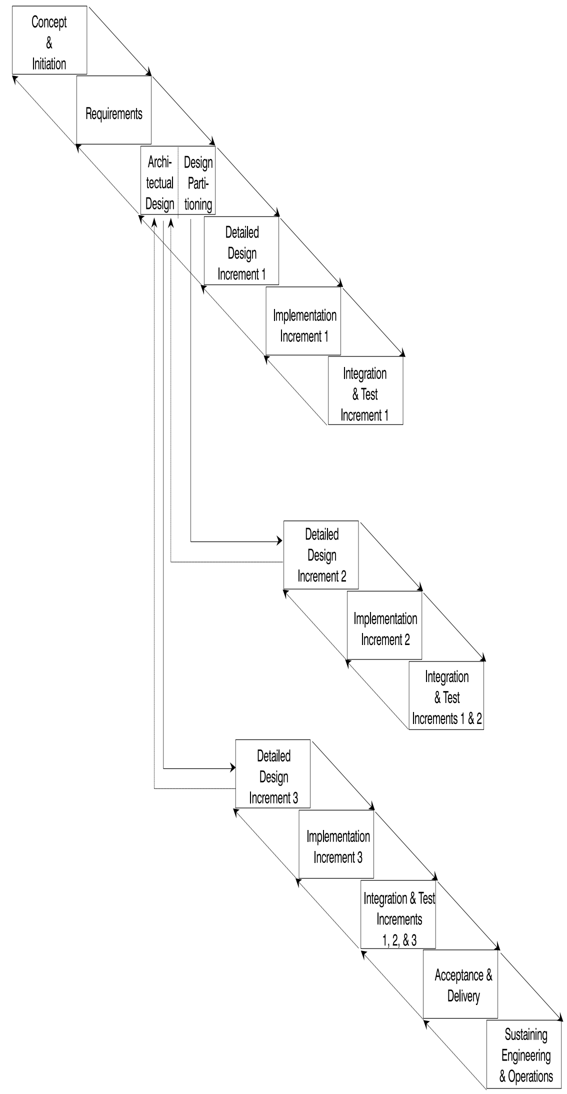


Figure 14-2. The Incremental Model

PHASED/EVOLUTIONARY MODEL
The Phased or Evolutionary model (Figure 3) plans for multiple iterations through the life cycle so that multiple deliveries of greater system capability are provided to the user. Each release or delivery incorporates the experience of the previous release and executes though all phases of the life cycle from Design through Sustaining Engineering and Operations. Iteration of part of a phase can occur for error correction.

This approach is best used for medium to high risk programs. Inherent in this design strategy is the concept that requirements and operational concepts will change as the user gains experience with the new system. It also depends heavily on user feedback to refine program requirements. From a technology perspective, it fosters the transition from "state-of-the-art" to "state-of-the-practice" by actively incorporating design features for the cost-effective integration of new technology as it matures. The criteria used to determine development progress and readiness for production and deployment are iterative in that it takes a "build a little, test a little, field a little" approach to rapidly put new capabilities into the hands of the user. This approach is best suited for either an evolutionary acquisition or preplanned product improvement acquisition strategy.

DESCRIPTION:
Phased approach with multiple iterations through the system development life cycle. Multiple releases of greater system capability are provided to the users. Each release incorporates the experience of previous releases and goes through all of the life cycle from requirements through deployment and operations. Each release can go through OT&E and full deployment.

SELECTION/REJECTION RATIONALE:
- Suitability for high technology and software intensive programs where requirements beyond the core capability can generally, but not specifically defined
- Responsive to changes in user requirements and technology maturity
- Focuses on functionality and cost effective delivery of functional performance to the user
- Enables periodic updates of existing functionality and delivery of new capabilities to the user based on user feedback
- A flexible, well planned overall architecture, to include process for change, which allows the system to be designed and implemented incrementally
- Early definition, funding, development, testing, deployment, and operational evaluation of an initial increment of operational capability
- Sequential definition, funding, development, testing, deployment, and operational evaluation of additional increments of operational capability

STRENGTHS:
- Early releases and functionality to users
- Users feedback can be incorporated in future releases
- Reduces risk through early demonstration of technical approach and requirements understanding, and partitioning of system into functional entities
- Re-usable library
- Prototyping used to demonstrate technical approach and promote user involvement in requirements definition
- Addresses the cost, risk, and relative time urgency of different elements of the system being developed
- Incorporates design features for cost effective incorporation of new technology/requirements
- Separate funding approval for each increment of operational capability, which should facilitate control of program costs

WEAKNESSES:
- Modification to operational releases may interfere with ongoing development
- Scheduling integration
- Test planning must be consistent with system partitioning and not focus on total system
- System partitioning highly dependent on ability to decouple functional entities

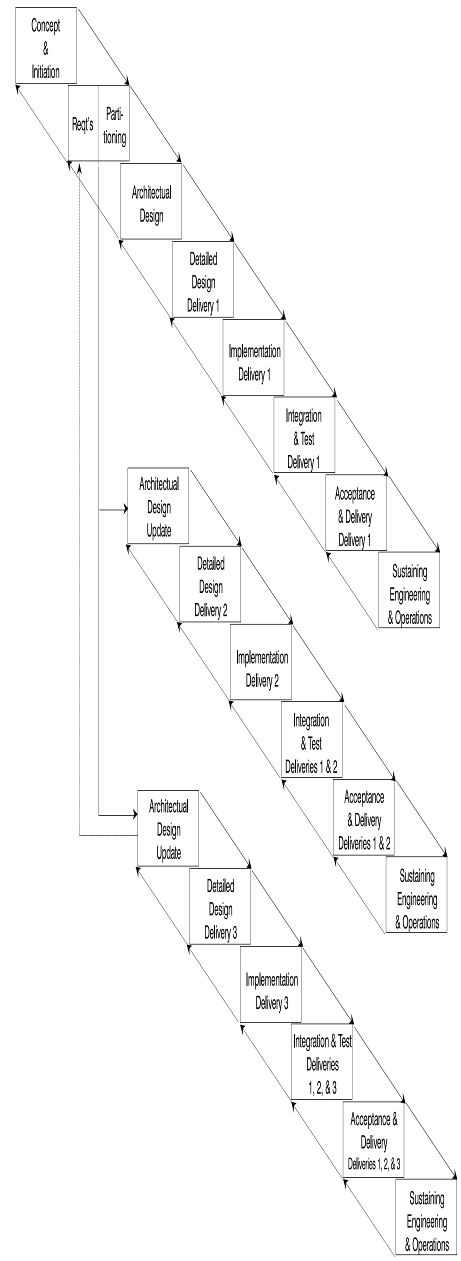


Figure 3. The Phased/Evolutionary Model.

SPIRAL/ITERATIVE MODEL

The Spiral or Iterative model (Figure 4) relies heavily upon the concept of prototyping. In this model, prototypes are planned throughout the development after initial requirements, after initial design, etc., so that an iterative cycle occurs with constant re-evaluations of the objectives, alternatives, and risks of the system development at each step. The extensive prototyping allows for the verification of portions of the overall system before continuing on to greater details of development. All of the life cycle phases defined in this section also occur in the Spiral model, however they are sequenced differently and may occur several times depending upon the evaluations of each prototype.

This is an approach where a core capability is fielded, and the system has a modular structure and provisions for future upgrades and changes as requirements are refined or become technically achievable. An evolutionary strategy is considered well suited to high technology and software intensive programs where requirements beyond the core capability can generally, but not specifically be defined.

DESCRIPTION:
Conceived by Barry Boehm in 1985 to emphasize the role of risk management and prototyping. Each phase of the spiral model involves:

- Selecting phase objectives
- Evaluating alternatives by means of prototyping, simulations, and modeling
- Determining remaining risks
- If risk low continue with waterfall approach

SELECTION/REJECTION RATIONALE:

- Limited application of processes and methods in a real-world environment
- Lacks methodology for selecting between alternatives using risk and opportunity as guides
- No provisions for periodic updates of existing functionality and delivery of new capabilities to the user based on user feedback
- Lacks an overall architecture, to include process for change, which allows the system to be designed and implemented incrementally
- Fragmented decisionmaking process makes it difficult to gain an overall perspective of the system and justify individual decision as being optimal

STRENGTHS:

- Provides a context in which system development and management personnel can cooperatively identify, examine, and resolve issues
- Identification of objectives gives a sense of direction to the project
- Repeated focus, within each cycle, on the objectives helps to keep the project from straying from the objectives
- Proactive approach of identifying and averting risks results in fewer surprises and reduced crisis management

WEAKNESSES:

- Gives the impression that risks are identified and averted in the same cycle
- Implies that risk is averted before development can proceed
- Methods on how to identify, develop, and test iterative or evolutionary builds are lacking
- The need for hierarchical and cooperating spirals - Their use, methods for coordinating with the higher level spiral, methods for tracking, and methods for reporting results need to be defined

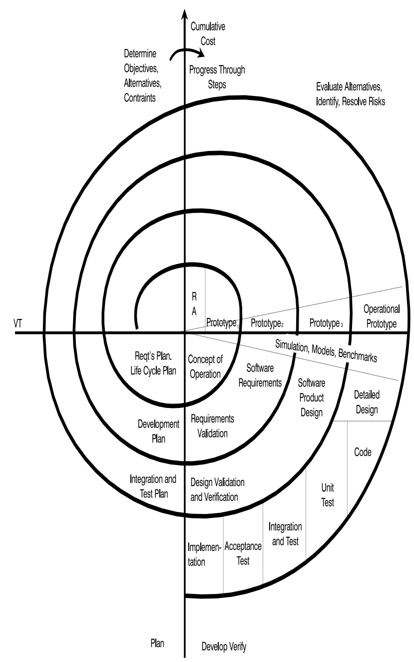


Figure 4. The Spiral/Iterative Model.

There are two basic contract types: **cost-reimbursement** and **fixed-price**. In a cost-reimbursement contract the government pays the incurred costs (subject to limitations on allowability, allocability, and reasonableness) and a fee that is either fixed or determined through a negotiated share formula which is used to adjust the fee. In fixed-price contracts, the government pays a negotiated price (cost and profit). In some cases; some fixed price contracts provide for an adjustable price that may include a ceiling price, a target price (including target cost and target profit) and a share formula.

There are two other "contract types", a **Blanket Purchase Agreement (BPA)** and **an Indefinite-Delivery, Indefinite-Quantity (IDIQ)**, that provide an overall ceiling price on the work to be performed and uses individual delivery/task orders that can be either fixed-price or cost-reimbursement to execute specific portions of the work.

The contract type determines how cost risk is shared with the contractor. As we move from cost-reimbursement toward fixed-price contracts, the contractor assumes more of the cost and performance risk. Properly structured, the appropriate contract type can provide effective incentives that address cost, schedule and performance risk.

Incentive contracting can be a powerful tool to challenge industry to superior performance. Pragmatic determination of strategy, sound selection, structuring of incentives, and careful management after award can enhance contractor performance. The type of incentive contract may require or permit varying management. Cost reimbursement contracts generally require more monitoring and administration; for example verifying incurred costs and ensuring that sufficient funds are obligated for the work.

Uncertainty is expected in incentive contracting. Cost/Price is typically incentivized. The pricing arrangement should provide for a fair profit/fee with reasonable increases or decreases in profit/fee for exceeding or failing to achieve incentive goals. However, if schedule or performance factors are important, additional incentives should be used. This can be done through inclusion of **objective** performance measures in incentive fee contracts or the use of **subjective** award fee contracts.

The incentive to control costs/price comes from the fact that the contractor's fee (if cost-reimbursement) or profit (if fixed-price) is adjusted when an overrun or underrun of the contract target cost occurs. The fee or profit adjustment is done through the application of a negotiated adjustment formula (commonly referred to as a share ratio). For example, if the contractor's actual cost exceeds the contract target cost, a cost overrun occurs. By applying the share ratio (see examples below) to the overrun, an amount is determined by which the contractor's target profit or fee is decreased. Should the contractor incur an underrun, the profit or fee would increase. Thus, the contractor is motivated to control costs because of the impact on profit or fee.

Share Ratio Examples

Award fee contracts with periodic subjective performance evaluations create a highly interactive relationship between the government and contractor. The government must be involved and understand the contractor's performance in order to have a basis for award fee determination. This interactive relationship requires more administration than the other contract types.

The contract type is an important decision that determines how risk is shared between the government and the contractor. Both fixed-price and cost reimbursement contracts can offer profit or fee as incentives for the contractor to control costs. There are tradeoffs involved in the choice of contract type that must be carefully weighed before the final solicitation package is released to industry. Within the **fixed-price** or **cost-reimbursement** categories we find the following common contract types:

<table>
<tr><td colspan="2">Fixed-Price Contracts – The contractor promises to deliver on time and meet the contract terms and conditions at a negotiated fixed price (regardless of how much it ultimately costs). This type of contract is used when requirements are well-defined, technology is mature, and sufficient historical data exists to permit accurate estimating and negotiation. Fixed price contracting is serious business. If a contractor is late, or his product doesn't meet the specifications for acceptance, the government has the right to terminate the contract for default and not pay the contractor for the cost of undelivered work. Clearly, in fixed price contracting the contractor bears a tremendous cost, schedule and performance risk.</td></tr>
<tr><td>Firm Fixed-Price (FFP)</td><td>In a FFP contract, price is the only element. The government pays a fixed price that is not subject to any adjustment, regardless of the contractor's cost experience. The contractor assumes all cost risk and has the greatest incentive to control costs. FFP contracts create a minimum amount of administrative effort for both parties since the contractor does not submit any cost data and the government does not need to verify the contractor's incurred cost. FFP contracts must be used when the risk involved is minimal or can be predicted with an acceptable degree of certainty. Clearly defined requirements and adequate cost data enable the negotiation of fair and reasonable contract prices prior to performance of any work.</td></tr>
<tr><td>Fixed-Price Incentive (FPI)</td><td>There are two types of FPI contracts: fixed-price incentive (firm target) (FPIF) and fixed-price incentive (successive target) (FPIS). Both contract types include a target cost which is the basis for determining the contractor's cost overrun or underrun. Using the overrun or underrun amount and the share ratio, the profit is adjusted upward or downward. A FPI contract also includes a ceiling price (cost plus profit) which is the maximum amount the government will pay regardless of cost incurred. If the contractor's cost exceeds the ceiling price, the contractor is in a loss position but must still deliver a quality product on time for the ceiling price. Though not as great as a firm fixed-price contract, the cost risk to the contractor is relatively high due to the cost limitation imposed by the ceiling price. The necessary elements for an FPI contract (negotiated prior to contract award) are:
Target Price/Schedule/Performance: best estimate of expected price/schedule/performance
Target Profit : profit contractor will receive if actual costs equal target price/schedule/performance
Share Ratio(s) : the formula used to adjust profit after actual costs are reviewed and accepted by the government
Ceiling Price : maximum amount the government will pay regardless of cost incurred</td></tr>
<tr><td>Fixed-Price-Plus-Award-Fee (FPAF)</td><td>FPAF contracts consist of a fixed price (cost and profit) and the "Award Fee Pool". The "Award Fee Pool" is a designated amount of funds that the contractor may earn based upon performance. The contractor will be paid the fixed price portion of the contract upon contract completion. For award fee purposes, the contract period of performance is divided into multiple periods of time with each period emphasizing different areas of performance on which the contractor will concentrate. The performance areas are negotiated by the government and contractor prior to contract award and delineated in the contract. The award fee funding is divided among the periods. For each period the contractor may earn 0%-to 100% of that period's award fee amount based on the government's subjective, unilateral determination of contractor performance.</td></tr>
<tr><td colspan="2">Cost-Reimbursement Contracts – The government promises to pay all allowable, allocable, and reasonable costs incurred on the contract. The contractor promises to exert its best efforts to perform the desired work. If more funds than originally estimated are needed, the contractor notifies the government which must provide additional funding or terminate the remaining work for convenience. Should the contractor continue working beyond the amount of funding provided after notification to the government, the government would incur an Anti-Deficiency Act violation if it does not order the contractor to stop work.</td></tr>
<tr><td>Cost-Plus-Fixed-Fee (CPFF)</td><td>In a CPFF contract, the fixed fee is negotiated prior to contract award and does not vary with actual costs. The government pays all allowable, allocable, and reasonable costs as well as the fixed fee. The cost risk to the contractor is minimal. However, this contract type provides the maximum amount of government risk</td></tr>
<tr><td>Time and Materials (T&M)</td><td>T&M contracts are a tailored version of a CPFF effort in that: the labor fee is embedded in fixed labor categories and rates that are projected over the life of the contract, and the material fee is fixed percentage of the material cost to the contractor that is indexed over the contract life. As with the CPFF, cost risk to the contractor is minimal and it provides the maximum amount of risk on the government.</td></tr>
</table>

Cost-Plus-Incentive-Fee (CPIF)	CPIF contracts are similar to FPI contracts in that both types include a target cost which is used to determine the contractor's cost overrun or underrun. Using the share ratio and the cost overrun or underrun, the contract fee is adjusted upward or downward. The difference between FPI and CPIF contracts is that the FPI contract limits the price the government will pay while the CPIF contract limits the amount of fee to be paid. The government will pay all allowable, allocable, and reasonable costs. However, a CPIF contract contains minimum and maximum fee amounts. Regardless of the cost overrun, the contractor will always receive at least the minimum fee. Conversely, regardless of the cost underrun, the contractor will only receive up to maximum fee. Thus, the contractor's cost risk is minimized because all their incurred costs are paid by the government and they are guaranteed the minimum fee. The necessary elements for a CPIF contract (negotiated prior to contract award) are: • Maximum Fee: greatest amount of fee that the contractor can earn • Minimum Fee: least amount of fee that the contractor can earn • Target Cost/Schedule/Performance: best estimate of expected cost/schedule/performance • Target Fee: fee the contractor will receive if actual cost equals target cost/schedule/performance • Share Ratio(s): the negotiated formula used to adjust fee after actual costs are reviewed and accepted by the government.
Cost-Plus-Award-Fee (CPAF)	CPAF contracts pay all allowable, allocable and reasonable costs. It also has two fee areas both of which are negotiated prior to contract award. The first area is known as the "Base" Fee (from 0 to 3%) which the contractor earns regardless of performance. The second area is the "Award Fee Pool" which is a designated amount of funds that the contractor may earn based upon performance. For award fee purposes, the contract period of performance is divided into multiple periods of time, with each period emphasizing different areas of performance on which the contractor will concentrate. The performance areas are negotiated by the government and contractor prior to contract award and delineated in the contract. Award fee funding is divided among the periods. For each period, the contractor may earn 0% to 100% of that period's award fee amount based on the **government's subjective, unilateral determination of contractor performance.**

Matching risk to the appropriate contract type is the basic key to narrowing in on the complex subject of selecting contract type. Expected costs of performance are usually uncertain, but our contracting approach should address "how" uncertain. If the work is clearly defined, the technology known, and there is historical or actual cost data available, the contract should probably be FFP, FPI, or FPAF. If the work cannot be adequately defined, the technology is state of the art and not yet fully matured, and there is inadequate historical or actual cost data available, then a CPFF, CPIF, or CPAF contract may be appropriate. Early in the developmental life cycle, cost type contracts are expected, with a move toward fixed-price contracts during production.

A comparison of the contract types discussed in this primer is presented on the next page. For additional information on contract types see:

Guidance on Contract Types and Incentives

Contract Pricing Reference Guides

http://www.acq.osd.mil/dpap/contractpricing/index.htm

Award and Incentive Fee Contracts CoP

https://acc.dau.mil/CommunityBrowser.aspx?id=105550&lang=en-US

Incentive Strategies for Defense Acquisitions

http://www.dau.mil/pubs/misc/INCENTIVE.pdf

Constructing Successful Business Relationships: Innovation in Contractual Incentives

http://www.acquisition.gov/comp/seven_steps/library/DOAconstructing.PDF

DOD and NASA Guide: Incentive Training Guide, 1969

https://acc.dau.mil/CommunityBrowser.aspx?id=189615&lang=en-US

Contract Types Primer

Contract Type	Principal Risk to be Mitigated	Use When . . .	Elements	Contractor is Obliged to	Contractor Incentive *(other than maximizing goodwill)*	Typical Application
Fixed-Price Contracts						
Firm Fixed-Price (FFP)	None. Thus, the contractor assumes all cost risk.	The requirement is well-defined. • Contractors are experienced in meeting it. • Market conditions are stable. • Financial risks are otherwise insignificant.	A firm-fixed-price for each line item or one or more groupings of line items.	Provide an acceptable deliverable at the time, place and price specified in the contract.	Generally realizes an additional dollar of profit for every dollar that costs are reduced.	Commercial supplies and services.
Fixed-Price Incentive (FPI)	Moderately uncertain contract labor or material requirements.	A ceiling price can be established that covers the most probable risks inherent in the nature of the work. The proposed profit sharing formula would motivate the contractor to control costs to and meet other objectives.	A ceiling price • Target cost • Target profit • Delivery, quality, and/or other performance targets (optional) • Profit sharing formula	Provide an acceptable deliverable at the time and place specified in the contract at or below the ceiling price.	Realizes a higher profit by completing the work below the ceiling price and/or by meeting objective performance targets	Production of a major system based on a prototype
Fixed-Price-Plus-Award-Fee (FPAF)	Risk that the user will not be fully satisfied because of judgmental acceptance criteria.	Judgmental standards can be fairly applied by the fee determining official. The potential fee is large enough to both: • Provide a meaningful incentive. • Justify related administrative burdens.	A firm fixed-price. • Standards for evaluating performance. • Procedures for calculating a fee based on performance against the standards	Perform at the time, place, and the price fixed in the contract	Generally realizes an additional dollar of profit for every dollar that costs are reduced; earns an additional fee for satisfying the performance standards.	Performance-based service contracts
Cost-Reimbursement Contracts						
Cost-Plus-Fixed-Fee (CPFF)	Highly uncertain and speculative labor hours, labor mix, and/or material requirements (and other things) necessary to perform the contract. The Government assumes the risks inherent in the contract - benefiting if the actual cost is lower than the expected cost-losing if the work cannot be completed within the expected cost of performance.	Relating fee to performance (e.g., to actual costs) would be unworkable or of marginal utility.	Target cost Fixed fee	Make a good faith effort to meet the Government's needs within the estimated cost in the Contract, Part I the Schedule, Section B Supplies or services and prices/costs.	Realizes a higher rate of return (i.e., fee divided by total cost) as total cost decreases.	Research study
Time and Materials (T&M)		No other type of contract is suitable (e.g., because costs are too low to justify an audit of the contractor's indirect expenses).	A ceiling price • A per-hour labor rate that also covers overhead and profit • Provisions for reimbursing direct material costs	Make a good faith effort to meet the Government's needs within the ceiling price.		Emergency repairs to heating plants and aircraft engines.
Cost-Plus-Incentive-Fee (CPIF)		An objective relationship can be established between the fee and such measures of performance as actual costs, delivery dates, performance benchmarks, and the like.	Target cost • Performance targets (optional) • A minimum, maximum, and target fee • A formula for adjusting fee based on actual costs and/or performance	Make a good faith effort to meet the Government's needs within the estimated cost in the Contract, Part I the Schedule, Section B Supplies or services and prices/costs.	Realizes a higher fee by completing the work at a lower cost and/or by meeting other objective performance targets.	Research and development of the prototype for a major system
Cost-Plus-Award-Fee (CPAF)		Objective incentive targets are not feasible for critical aspects of performance. Judgmental standards can be fairly applied. Potential fee would provide a meaningful incentive.	Target cost • Standards for evaluating performance • A base and maximum fee • Procedures for adjusting fee, based on performance against the standards	Make a good faith effort to meet the Government's needs within the estimated cost in the Contract, Part I the Schedule, Section B Supplies or services and prices/costs.	Realizes a higher fee by meeting judgmental performance standards.	Large scale research study

Program Execution Oversight

A quantitative assessment structure is driven by the information needed to address management issues. Specific information needs and concerns determine the selection and application of attribute measures and the analysis focus. These can and will change over the program's life cycle. The overall measurement process is intended to support the immediate and projected information needs of the program manager. Information needs are categorized as program, process, product, and risk. The data, indicators, and analysis results support the questions defined below and provide for objective communications between the acquirer and the developer.

Program Oriented Questions. Program planning, execution, and status are of key interest to the program manager. Program plans, e.g., Systems Engineering Management Plan, Software Development Plan, Continuous Process Improvement Plan, etc., establish process whose purpose is to satisfying the technical and operational performance objectives of the development effort. Execution translates the plan into actual processes being used and refines the measures of merit that will be used to validate the achievement of performance (quality) and technical objectives. Status provides the program manager with the current status in the development process and related product quality, and the ability to predict when the program will be completed based upon quantitative evidence. Sample program questions include:

Question	Basis of Answer
What is the status of the program?	This question will be answered in terms of work breakdown structure (WBS), schedule, and the resource plan/action.
For what purpose is the program intended?	The answer will address requirements traceability, requirements understanding, and end product validation.
What are the customer's needs?	This will consider the urgency associated with satisfying the customer's needs and the identification of the appropriate product quality factors allied with the requirements.
How stable are the user's requirements?	The question will be answered using the results of the requirements analysis process, including the determination of stated and derived requirements, affects the translation of these requirements into design and product specifications.
Is there early identification of user requirements and changes?	Related to the requirements analysis process, this question focuses on customer needs, both stated and derived, and the change control process that can affect program development and customer satisfaction.
Is the product mature enough to be used in operation?	Maturity will be addressed in terms of product quality metrics, readiness for test and evaluation, and readiness for fielding and/or deployment.

Process Oriented Questions. The process defines the set of activities that will be performed to produce the desired products. The program manager is typically interested in the resources associated with the process (including cost, schedule, and resources) and how the process is impacted by scope and requirements changes. Sample process questions include:

Question	Basis of Answer
Are costs under control?	Funds availability, estimated costs to completion, and tracking costs against planned expenditure rates form the key elements of cost control.
Is the project on schedule?	Schedule aspects may be tracked at the upper level and specific activity level, e.g., design, code, test.
Is the project within computer resource limitations?	Computer resource limitations can apply to memory, throughput, input/output constraints, etc.
Are scope changes impacting the project?	Potential scope changes include any change to the deliverable products and tasks contained in the SOW.
Are requirement changes impacting the project?	Requirement changes include those presented by the user and those derived during the requirements analysis and allocation process.
Is the installation and/or user ready to receive the product?	Have the necessary physical, workflow, and organizational modifications been identified and made to use the product?

Product Oriented Questions. All development and sustaining engineering activities produce products. Products include, but are not limited to, designs, documentation, executable code, prototypes, software systems, source code, and specifications. The program manager is interested in the quality of these products. Sample product questions include:

Question	Basis of Answer
Are defects being identified and corrected?	Defect tracking can provide insight into the source of defects, defect discovery rates, and defect closure rates.
Is the product of high quality and is the end user satisfied?	With quality being defined in terms of one or more of the following quality attributes: maintainability, correctness, reusability, testability, portability, reliability, adaptability, availability, transportability, flexibility, and/or safety.

Risk Oriented Questions. Identifying, assessing, and managing risk is a key program management function. As visibility increases into the development process and the resultant product quality, program management can make maximum use of collected data to support system risk management. To support this function, typical risk related questions include:

Question	Basis of Answer
What is the technical risk?	Technical risk is the cornerstone of risk control and affects the inherent risks for all other risk areas. Technical risk control includes determination of the performance envelope of the development and operational software within the context of the overall system.
What is the cost risk?	Cost estimates, normally deterministic, are fundamental to program management. Costs for individual program elements are usually expressed as a single value representing the "best" available estimate.
What is the schedule risk?	Schedule risk control includes determination of the range of times covering the span between earliest and latest dates for program completion, with detailed schedules and risks
What is the operational risk?	Using a range of estimates, identifying the most likely outcome, and assessing the risk of having a system that does not meet requirements in the customer's operational environment are all components of operational risk control.
What is the support risk?	There are several issues which complicate risk control in the sustaining engineering (support) environment. Among these issues are: determination of what facilities, personnel, and procedures are necessary for sustaining engineering; and what re-sources are available for sustaining engineering.

Metrics for Answering the Questions

The measurement process should be flexible and tailorable to distinct development program characteristics, objectives, and limitations. It should be adapted to different development process models and implementations. Once the measurement process is tailored to meet the specific information needs of an individual program, it should be consistently applied throughout the development or sustaining engineering process.

Attribute measures are not constrictive. Within the overall measurement process, different measures can be applied for different programs. The integrity of the metrics implementation with respect to a particular development application ensures valid and objective analysis results.

Attribute measurement methodologies should be well defined and consistently applied across the development. The actual measures applied during the development change with respect to changing development process activities, development products, and information requirements. Measurement should be implemented at the lowest practical level. Multiple measures should be applied with respect to a given issue to support cross-validation of the measurement findings.

Generic information categories (process characteristics, product characteristics, product quality, progress, and resources) that can be used to answer the above questions are presented in Table 2. Since these information categories are multi-dimensional, metric categories that can be used to address the specific attributes of each information category are also identified. These information categories can, in turn be related to the metric categories and individual metrics that can supply the necessary data that through the data analysis process provide the necessary information to meet the decisionmaker's information needs.

INFORMATION CATEGORIES	**Process Characteristics:**	**Product Characteristics:**	**Product Quality:**	**Progress:**	**Resources:**
ATTRIBUTES	Applicable Metric Categories				
Complexity		Size & Complexity			
Configuration	Stability	Stability	Stability		
Cost	Management			Management	Management
Defects		Testing	Testing		

Earned Value		Stability		Stability	
Environment	Management			Management	Management
Productivity	Management		Testing	Management	Management
Progress	Stability	Stability	Testing	Stability	
Quality		Features	Features		
Requirements	Stability	Stability	Stability		
Schedule	Management			Management	Management
Size	Size & Complexity	Size & Complexity	Size & Complexity	Size & Complexity	
Stability	Stability	Stability	Stability	Stability	Stability
Staffing	Management		Management	Management	Management
Test	Testing	Testing	Testing	Testing	

Table 2. Multi-Dimensional Information Requirements Are Satisfied by Metrics.

Just as the information categories are multi-dimensional, so is the relationship of the information categories to the various project, process, product, and risk questions. This relationship is shown in Table 3. The multi-dimensionality of the information categories and the answers to the questions provides the different perspectives necessary to prevent misinterpretation of a single metric or metric category. This built in cross-checking also promotes a return on investment of the metrics data collection costs by making maximum use of the collected data. It also provides a structure that present the information in terms that are useful in the management, e.g., oversight, program, engineering, etc., decisionmaking process and is not too detailed or too sketchy so that it doesn't lose its value to the decisionmaker..

Measurement Goal: To provide decisionmakers with the necessary information that supports the project's productivity, quality, process improvement, and risk management objectives.					
	Information Requirements				
Questions	**Process Characteristics**	**Product Characteristics**	**Product Quality**	**Progress**	**Resources**
Project Questions					
1. What is the status of my program?	X	X	X	X	X
2. For what purpose is the program intended?		X			
3. What are the customer's needs?		X			
4. How stable are the user's requirements?		X			
5. What are the risks to my program?	X	X	X	X	X
6. Is there early identification of user requirements and changes?		X			
7. Is the product mature enough to be used?		X	X	X	
8. When do I freeze my requirements and/or design?	X	X	X	X	X
Process Questions					
1. Are costs under control?	X			X	X
2. Is the project on schedule?	X	X	X	X	X
3. Is the project within resource limitations?		X			X
4. Are scope changes impacting the project?		X		X	X
5. Are requirement changes impacting the project?		X		X	X
Product Questions					
1. Are defects being identified and corrected?		X	X	X	
2. Is the product of high quality?		X	X	X	
Risk Questions					
1. What is the technical risk?		X	X		
2. What is the cost risk?	X			X	X
3. What is the schedule risk?	X			X	X
4. What is the operational risk?		X	X		
5. What is the support risk?		X	X		X

Table 3. Information Categories Related to the Program Manager's Questions

Figure 1 identifies specific activities that can be conducted during any phase of the life cycle. These activities can be grouped into six top level activities: requirements analysis, design, implementation, integration and test, support and delivery, and integral and common activities. The generic metric types can be directly related to the top level activities, with specific data primitives that can be collected during each activity. Data primitives can then be input into specific metrics that are chosen by the program office to meet management's information needs. The chosen metrics can be used to provide the feedback and status information necessary to continuously monitor and improve the process and products.

TOP LEVEL ACTIVITIES	**REQUIREMENTS ANALYSIS**	**DESIGN**	**IMPLEMENTATION**	**INTEGRATION & TEST**	**DELIVERY and SUPPORT**
Activities	System requirements analysis Hardware/Software requirements analysis	System design Hardware design Software design	Hardware/Software implementation and testing	Unit integration and testing CI Qualification testing CI Integration and testing System qualification testing	Preparation for use Preparation for transition
Integral & Common Activities	Project planning Development environment Product evaluations Configuration management Joint technical and management reviews Corrective action Quality assurance Other management activities				
INFORMATION CATEGORIES	**APPLICABLE METRIC CATEGORIES**				
Process	Management		Size and Complexity	Stability	Testing
Product	Management	Features	Size and Complexity	Stability	Testing
Product Quality	Management	Features	Size and Complexity	Stability	Testing
Progress	Management		Size and Complexity	Stability	Testing
Resources	Management	Features	Size and Complexity	Stability	

Figure 1. Top Level Activities and Applicable Metric Categories

Objective evaluation of development processes and products uses a quantitative development assessment structure. The assessment process is based upon quantitative measures (metrics) of process and product attributes applied across the life cycle. These attributes are evaluated within the assessment structure with respect to program characteristics, constraints, and objectives, and integrated into an overall assessment of the process and resultant products within the overall context of the program.

The program manager's information needs are satisfied by multiple disciplines. The metrics used provide the information that can be used by these disciplines in a form that is compatible with the normal information analysis methods and tools used by these disciplines (Table 4). Use of product information by these other disciplines fosters the awareness of product impacts on these disciplines and promotes the presentation of the product information within a system context.

DISCIPLINES	**ACTIVITIES**	**APPLICABLE METRICS**
POLICY AND ORGANIZATIONAL MANAGEMENT		
ACQUISITION POLICY	ACQUISITION PHASES ACQUISITION STRATEGY DECISION POINTS	MANAGEMENT STABILITY SIZE and COMPLEXITY TESTING
PRINCIPLES OF PROGRAM MANAGEMENT	PLANNING ORGANIZING AND STAFFING CONTROLLING AND EXECUTING	MANAGEMENT STABILITY
MANAGERIAL DEVELOPMENT	INTERPERSONAL COMMUNICATION PROGRAM MANAGEMENT TEAM PERSONNEL MANAGEMENT	MANAGEMENT
BUSINESS MANAGEMENT		
CONTRACTOR FINANCE	FINANCIAL ANALYSIS AND FORECASTING FINANCIAL REPORTING AND COST MANAGEMENT CONTRACT PRICING	MANAGEMENT STABILITY
COST/SCHEDULE MANAGEMENT	CONTRACT FUNDS STATUS REPORTING COST PERFORMANCE REPORTING PROGRAM LIFE CYCLE COST ESTIMATE WORK BREAKDOWN STRUCTURE	
CONTRACT MANAGEMENT	CONTRACT TYPE SELECTION RFP PREPARATION	

	SOURCE SELECTION	
	CONTRACT EXECUTION	
TECHNICAL MANAGEMENT		
SYSTEMS ENGINEERING	SYSTEM REQUIREMENTS ANALYSIS	MANAGEMENT FEATURES STABILITY SIZE AND COMPLEXITY TESTING
	CONFIGURATION MANAGEMENT	
	DATA MANAGEMENT	
	SYSTEM DESIGN	
	SYSTEM INTEGRATION	
	SYSTEM PERFORMANCE ASSESSMENT	
SOFTWARE MANAGEMENT	SOFTWARE REQUIREMENTS ANALYSIS	
	SOFTWARE DESIGN	
	SOFTWARE INTEGRATION	
	COMPUTER RESOURCES	
	SOFTWARE PERFORMANCE ASSESSMENT	
TEST AND EVALUATION	DEVELOPMENT TEST	
	OPERATIONAL TEST	
LOGISTICS SUPPORT	LOGISTICS SUPPORT ANALYSIS	
	SUPPORT CONSTRAINTS	
	POST PRODUCTION SUPPORT	

Table 4. Metrics Can Be Used By Multiple Disciplines.

Statistical Based Testing Primer

Introduction

Test and Evaluation (T&E) is an integral part of the Systems Engineering Process which identifies levels of performance and assists the development in correcting deficiencies. Statistical Based Testing is a tool used by T&E professionals to determine sample sizes for testing, establish confidence levels and confidence intervals for the test results, and improve the overall efficiency and effectiveness of T&E. A key function of a test experiment is to determine the statistical confidence associated with the performance of an item under test. A classical approach to conduct this T&E process is using the guideline in the Military Standard 1916, Department of Defense (DoD) Preferred Methods for Acceptance of Product (MIL-STD-1916). DoD has adopted the MIL-STD-1916 sampling plans instead of the traditional MIL-STD-105E, Sampling Procedures and Tables For Inspection By Attributes, which was the military standards for over 30 years.

Design of Experiments (DOE) is a scientific and statistical principle for Test and Evaluation. DOE serves as a structured process to assist in developing T&E strategies utilizing statistical analyses. It develops an integrated operational test program by providing confidence to a system performance. DOE is an efficient procedure for planning experiments so that the data obtained can be analyzed to yield valid and objective conclusions. JITC has adopted JMP to be the DOE analyses tool.

Below are some of the key definitions for both the Classical and DOE approach.

Classical Approach (MIL-STD-1916)	DOE Approach (JMP Tool)
Verification Level (VL) is the level of significance to the user. The amount of effort to assure conformance can be allocated on the basis of importance to the user. Level I to VII, where VII requests the highest level of effort. Similar to Acceptable Quality Level (AQL)	**Confidence Level** is a percentage representing the margin of error in an experiment. (Criteria includes: margin by which the specifications were exceeded or missed, number of trails, and amount of data variability)
Code Letter (CL) is a letter representing the lot or production interval size based on the verification level	
Average Outgoing Quality (AOQ) is the average quality of outgoing product of non-defective sample set.	
Lot Tolerance Percent Defective (LTPD) is a designated high defect level that is unacceptable.	**Confidence Interval** is the acceptable margin of error.
Defect is any non-conformance of the unit of product with specified requirements.	**Difference to Detect** is the value of how small a difference you want to be able to declare statistically significant to test against.
Defective is a unit of product that contains a defect.	
Lot or Batch is a collection of items from which a sample will be drawn.	**Population Size** is the total number of people/test case/test items/etc that are in the group the sample represents.
Process Average is the average number of defects per hundred units of product submitted by the original inspection.	
Sample is one or more units of product from a lot or batch. It is selected at random.	
Sampling Plan is a sampling scheme and a set of rules for executing a test experiment. The result of executing a sampling plan is to determine a decision that can be to	

Classical Approach (MIL-STD-1916)	DOE Approach (JMP Tool)
accept or reject the lot.	
Unit of Product is the product inspected.	**Factor** is the entity whose effect on the response is investigated.
Mean is the average of the data.	
Variability is the measure of scatter of data in terms of variance or standard deviation. (The smaller the variance, the more confident in resultant conclusion.)	
Sample Size is the number of items that is randomly chosen or calculated from a population. It's calculated based on three factors: Confidence Interval, Confidence Level, and Population.	**Sample Size** is the total number of observations in the experiment.
Single Sampling is a sample of a group randomly selected from a lot/batch to determine from the resulting information.	
Double Sampling occurs after a modified (smaller sample size) single sampling's result has be rejected. Typically the 2nd sampling conducts a larger sample size to combine with the results of the first sampling.	
	Power is the probability of getting significance differences between two groups at a given significance level. The higher the power value, the bigger the sample size is required.
	Alpha Level is the probability of rejecting the null prediction. It is also referred to as a "Significance Level"
	Standard Deviation (Std Dev) is a measure of the unexplained random variation around the mean.

Classical Based Test Approach

MIL-STD-1916 emphasizes on an effective prevention based strategy quality system. The sampling plan is based on a "zero accept one reject" as criteria of judgment. Also, the sampling tables are less than those of MIL-STD-105E which makes it simpler and user friendly. However, the MIL-STD-1916 sampling plans are not intended for use with destructive test experiments or where product screening is not feasible.

MIL-STD-1916 for Attributes is one of the classical approaches to statistical based testing. By using the sampling procedures and tables for attributes provided in the MIL-STD-1916 standards, conducting the experiment will include:

- Defining a Population/Lot Size for the experiment.
- Choosing a Verification Level.
- Determining the appropriate sampling plan to be conducted with the experiment.

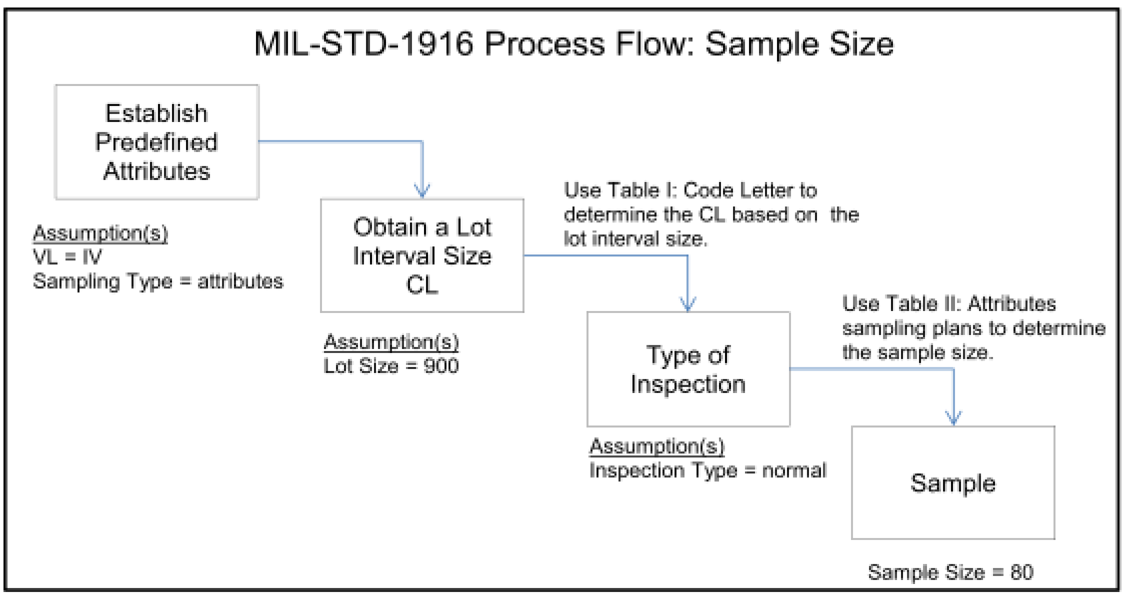
The Type of Inspection for MIL-STD-1916 determines the level of experiment. There are three types of inspections:

- **Normal** inspection is used at the start of the inspection activity.
- **Tightened** inspection is used when the recent quality history has deteriorated. (acceptance criteria are more stringent than under normal inspection)
- **Reduced** inspection is used when the recent quality history has been improved. (sample sizes are usually smaller than under normal inspection)

There are quick and simple alternatives to calculate the sample size needed in order to get results that reflect a given target's population. The site http://www.surveysystem.com/sscalc.htm provides a straightforward user interface to calculate the sample size needed when given the input parameters Confidence Level (95% or 99%), Confidence Interval, and Population. The site also can calculate the confidence Interval when the input parameters are Confidence Level, Sample Size, Population, and Percentage. The Society for Quality Control (SQC) online site < http://www.sqconline.com/mil-std-1916-calculator-acceptance-zero-beta?AVC=A > calculates the sampling size by attributes, in accordance to MIL-STD-1916, for a given lot size, verification level and the Type of Inspection.

The overall process for MIL-STD-1916 is illustrated in the diagram below. The next paragraph describes an example of the process. MIL-STD-1916 begins by establishing a Verification Level (VL) of a sampling plan. Next step is to determine the type of sampling which consist of attributes, variables, and continuous samplings. Follow by identifying the Code Letter (CL) from Table I based on the lot interval size. After acquiring the CL, the type of inspection for the experiment needs to be selected (normal, tighten, reduced). The last step is to review the sampling plan reference table to acquire the sampling size for the predefined attributes.

The example below demonstrates a basic classical based test approach using MIL-STD-1916. First, we defined the assumptions where the VL is "*IV*", the type of sampling is "*Attribute*" sampling, and the Lot/Population Size is "900". Afterwards, the CL is determined to be "A" by using the MIL-STD-1916 table I: "Code letters (CL) for entry into the sampling tables". Next we assume the Inspection Type for this experiment is *"Normal"*. The last step is to acquire the sample size from MIL-STD-1916 Table II: "Attributes sampling plans". Base on the attributes selected, VL, and CL, the sample size for this experiment is "80".



DOE Approach

DOE is a process to develop an experimentation strategy to maximize gaining knowledge of using a minimum amount of resources. DOE methodology ensures all factors within the experiment are inspected. There are a few benefits for using the DOE approach: 1) Testing multiple factors to separate the critical factors from the insignificant factors. 2) Evaluating the effects of factor interactions that can be used to optimize performance and to predict outcomes. 3) Confirming a system that performs within its requirements.

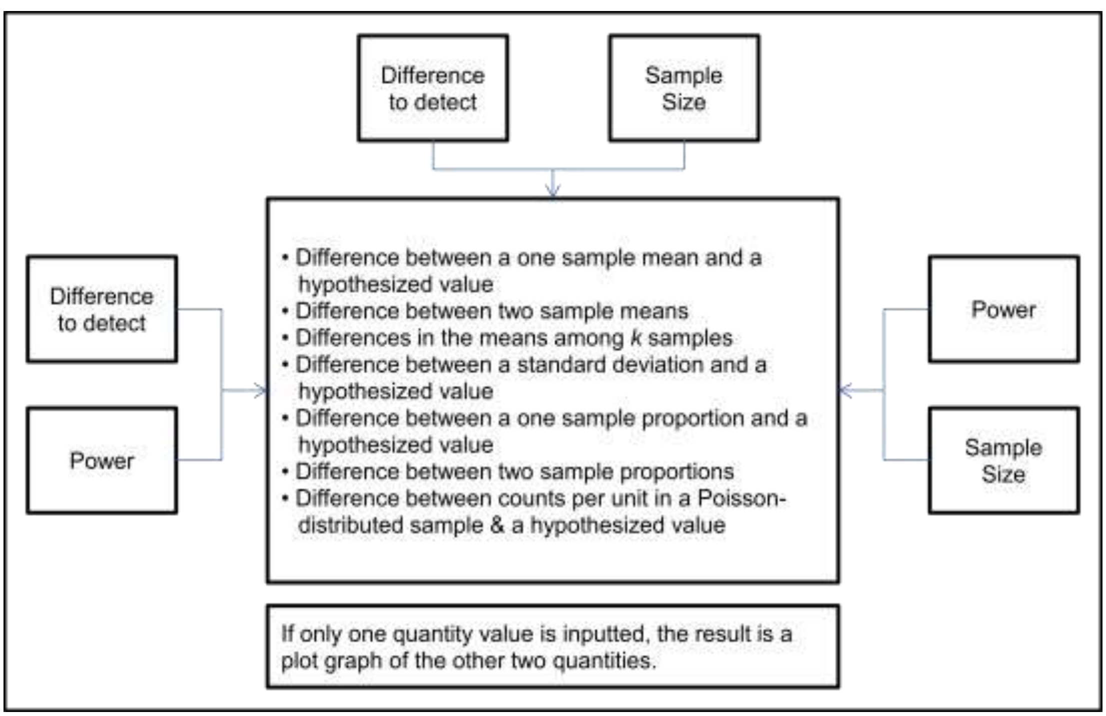
There are four key phases for conducting a DOE experiment:

- Phase 1: Plan – A statement of the experiment goal and develop a list of variables related in the experiment.
- Phase 2: Design – Select a design type to determine the number of tests to be conducted in the experiment.
- Phase 3: Execute – Perform each test to be executed in the experiment.
- Phase 4: Analyze – Mathematical study of the resulting data to report the conclusion.

JMP is a product of SAS Institute and was designed around a useful graphical user interface (GUI). The tool is a desktop statistical software that provides an effective visualization techniques for data recovery and analysis. In-depth documentation/books are available for purchase. Designing experiments in JMP is centered on factors, responses, a model, and runs. JMP helps to determine if and how a factor affects a response.

The JMP tool does not calculate a sample size based on these three factors. An alternative is to calculate the sample size by using another tool. Once the baseline of the sample size is determined, JMP tool can improve the sample size value by using different type of comparison calculations in the analysis.

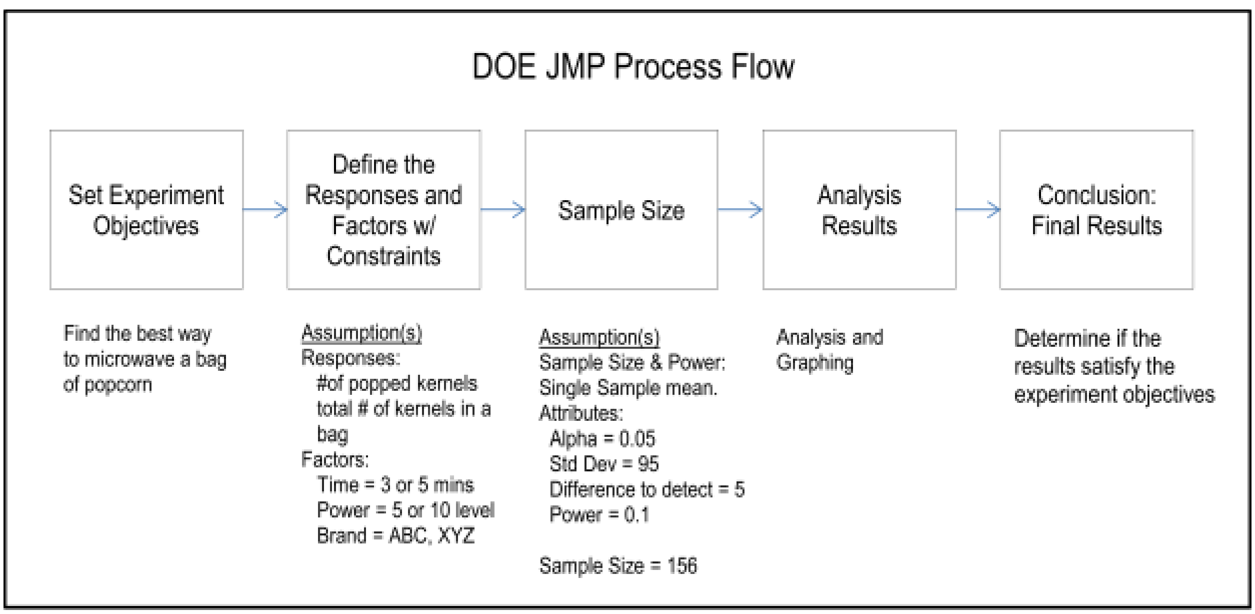
The functionality "Sample Size and Power" in the DOE JMP tool is to answer a key question of "how many runs does it take?" It's a platform that helps you plan your study for different type of comparisons. The results of the JMP tool depend heavily on the different available input parameters. Required sample size, expected power, and expected effect size are the important quantities for the calculation. These quantities depend on the significance level of the hypothesis test for the effect. Depending on the experimental situation, one or two quantities are supplied to obtain a third quality. If one value of the quantity is supplied, the result is a plot of the other two. And if two quantity values are supplied, the third value is calculated. Below illustrates the possible calculation when two quantity values are provided.



The following summarize the different experimental situations available:

- One Sample Mean - Sample Size for testing a mean in a single sample
- Two Sample Mean - Testing that the means are different across 2 samples
- k Sample Means - testing that the means are different across k samples
- One Sample Standard Deviation - Sample Size for detecting a change in the standard deviation
- One Sample proportion - Sample Size for testing a proportion in a single sample
- Two Sample Proportions - Sample Size for testing a proportion across 2 samples
- Counts per Unit - Sample Size for detecting change in count per unit, e.g. DPU (defects per unit)
- Sigma Quality Level - Calculator for a popular index in terms of defects per opportunity
- Reliability Test Plan - Sample size for reliability studies
- Reliability Demonstration - Calculations for planning a reliability demonstration.

DOE in JMP focuses on factors, responses, a model, and runs. The following diagram demonstrates an experiment using the JMP tool. The first step is to determine the experiment objectives. The objective for this example is finding the best way to microwave a bag of popcorn. Once the objective is defined, the JMP tool uses a custom designer to setup the DOE design by inputting the Responses and Factors, including constraints, for the experiment. The responses selected are “number of popped kernels” and “total number of kernels in a bag”. The factors and factor constraints are “time = 3 or 5 minutes”, power = 5 or 10 level”, and “brand = ABC, XYZ”. Next is to define the Sample Size needed to perform the experiment with the effects in the model by using the “Sample Size and Power” functionality. For this step, we calculate the Sample Size to be 156 by using the Single Sample Mean to calculator and setting the attributes “Alpha = 0.05”, “Std Dev = 95”, “Difference to detect = 5”, and “Power = 0.1”. After reviewing the design, execute the experiment, gather and enter the data into JMP. The last step is to analyze the data using graphical and different visualization plots to report results.



Comparison

The Classical Statistical and DOE approaches have its advantages.   Determining a statistical approach to apply in an experiment is situational. The table below shows the key benefits and differences for each of the approach.

Benefits and Differences	
Classical Statistic Approach	**DOE Approach**
For large scale high volume production products	Ability to analyze large amount of data (transform data and create theoretical model for analyzing)
Can implement unique scenario	Scalability (able to work with small and large services)
A one-shot execution sampling plan experiment. (test short-run effect)	Choosing between alternatives by performing Comparative Experiment.
	Determining key factors by performing Screen Experiment.
	Ensure multiple factors and their interactions are examined.

References

- NIST Engineering Statistics Handbook
  < http://www.itl.nist.gov/div898/handbook/index.htm >
- SQC Online Calculations for various Acceptance sampling
  < http://www.sqconline.com >
- Weibull Experiment Design and Analysis Reference
  < http://www.weibull.com/doewebcontents.htm >
- MIL-STD-105E Documentation, 10 May 1989
  < http://archive.org/details/MIL-STD-105E_1>
- MIL-STD-1916 Documentation, 01 April 1996
  < https://acc.dau.mil/CommunityBrowser.aspx?id=523195 >
- JMP Reference Documentation
  < http://www.jmp.com/support/downloads/documentation.shtml >

RAM Primer

Primary Source: DoD 3235.1-H, "Test and Evaluation of System Reliability, Availability, and Maintainability - A Primer"

RAM Considerations

There is a program management training short course that goes: up-front, early-on; be proactive; and it depends. These same elementary concepts also apply to the practice of RAM principles and can be a major contributor to the success or failure of a system's RAM program. In particular, the following activities are crucial to successful RAM efforts:

1. **Define**. Up-front, it is crucial that measures to be used (possibly tailored) and their input parameters (e.g., duration, failure, fault, critical, etc.) be defined to ensure data consistency, uniform information interpretation, and value-added reporting. For example, many of the measures in this primer use time as the basic representation of duration, however, for a particular system, time may be irrelevant. A better representation of duration may be events. As long as duration is defined and agreed to by all the relevant parties, using a different duration representation should not impact the overall usefulness of the RAM measure.

2. **Count**. Early-on, in addition to defining the data to be collected and how it will be used to calculate the appropriate RAM measure, it is essential to determine: what is to be counted in the analysis, when the data is to be collected, and how data from different life-cycle phases will be used. For example, the contract for a software-intensive system included a reliability incentive. The contract stated that if a software failure occurred and it was corrected within a specified time period, the failure would only be counted once. If the software failure was not corrected within the specified time period, each failure occurrence would be included in the reliability calculation.

3. **Corrective Actions**. It pays to be pro-active in defining how the ramifications of corrective actions will be addressed in calculating and presenting the RAM measures. In particular, pay attention to: the validity of applying previously collected data to the new system baseline (post corrective action), isolating new contributors to overall system RAM performance, and the use of RAM measures to incentivize performance.

4. **Reality**. Many measures in this primer reflect a binary world of either it works or it doesn't. Reality is much more of a "it depends" world. For example: a system can be up, but degraded, and provide mission capabilities that otherwise would not be available to some, but not necessarily all, users; a workaround could be available to readily overcome a condition that otherwise would be considered a failure; or a failure external to the system could be giving the appearance of a system fault. These considerations, and others, should be taken into consideration when establishing a RAM program.

DEFINITIONS

ALDT	Administrative and logistics down time spent waiting for parts, administrative processing, maintenance personnel, or transportation per specified period (Delay-down time).
MDT	Mean down time.
MTBF	Mean time between failures.
MTBLOF	Mean time between loss of function
MTBM	Mean time between maintenance actions
MTBUMA	Mean time between unscheduled (corrective) maintenance actions.
MTTR	Mean time to repair.
MTTRF	Mean time to restore function.
OMF	Operational mission failure
OT	Operating time (system in use).
ST	Standby time (not operating, but assumed operable) in a specified period.
TCM	Total corrective (unscheduled) maintenance time per specified period.
TDT	Total down time = TMT + ALDT.
TMT	Total maintenance time = TCM + TPM (Active-down time).
TPM	Total preventative (scheduled) maintenance time per specified period.
TT	Total intended utilization period, total time.

Reliability is the ability of a system and its parts to perform its mission without failure, degradation, or demand on the support system. Reliability is multifaceted and has a number of different aspects to be considered. Reliability can be defined in terms of several different non-mathematical and mathematical expressions, all of which depend upon long duration exposure to use (Table 1).

TYPE	PURPOSE	MATHEMATICAL EXPRESSION
Mission Reliability	Used to address the probability that a system will perform its required mission critical functions for the duration of a specified mission under conditions stated in the mission profile (or operational scenario).	Mission reliability can also be stated as the probability a system can complete its required operational mission without an operational mission failure (OMF)
Logistics Reliability	Considers all calls for the use of logistics resources, not taking into the measure of the ability of an item to operate without placing a demand on the logistics support structure for repair or adjustment.	Measures of logistics reliability include the probability that no corrective (or unscheduled) maintenance, unscheduled removals, or unscheduled demands for spare parts will occur following the completion of a specific operational scenario.
Mean Time Between Failure (MTBF)	Used to get a measure of the impact of all failures, regardless of type, on the overall ability of the system to perform its mission.	$MTBF = \frac{OT}{\text{Number of Failures}}$
Mean Time Between Operational Mission Failures (MTBOMF)	Used to represent the average time experience between failures that prevent the system from performing one or more mission essential functions	$MTBOMF = \frac{\text{Total operating time (or system-on time)}}{\text{Total number of operational mission failures}}$
Mean Time Between Mission-Critical Failure (MTBMCF)	Similar to MTBOMF, but addresses those failure that prevent a system from performing its intended mission, e.g., a mission abort for a strike aircraft.	$MTBMCF = \frac{\text{Total operating time (or system-on time)}}{\text{Total number of mission critical failures}}$

TABLE 1. Reliability Expressions.

Availability translates system reliability and maintainability characteristics into an effectiveness index. Availability measures the degree to which an item is in an operable and committable state at the start of a mission when the mission is called for at a random point in time. Availability can be defined in terms of several different mathematical expressions, all of which depend upon long duration exposure to use (Table 2).

TYPE	PURPOSE	MATHEMATICAL EXPRESSION
Basic (A)	Used to define system availability in terms of up time and down time.	$A = \frac{\text{Up Time}}{\text{Total Time}} = \frac{\text{Up Time}}{\text{Up Time + Down Time}}$
Inherent Availability (A_i)	Used to define system availability with respect to operating time and corrective maintenance.	$A_i = \frac{MTBF}{MTBF + MTTR}$
Operational Availability (A_o)	Covers all segments of time that the system is intended to be operational.	$A_o = \frac{OT + ST}{OT + ST + TPM + TCM + ALDT}$
Achieved Availability (A_a)	Used during development test and initial production test when the system is not operating in its intended support environment.	$A_a = \frac{OT}{OT + TCM + TPM}$
Modified Inherent Availability (A_{im})	Used to address multi-mission and/or multi-mode systems that enable the system to "automatically" correct for a component failure, e.g., automatic switching, or continue to operate in a degraded, but still functional mode.	$A_{im} = \frac{MTBLOF}{MTBLOF + MTTRF}$

TABLE 2. Mathematical Expressions of Availability.

For multi-mission and/or multi-mode systems, it is necessary to determine up and down times as a function of each mission/mode. This generally requires using separate time line models for each identifiable mission/mode. Likewise, separate time line models are generally required to support availability analysis of systems that experience significantly different peacetime, sustained combat, and surge utilization rates.

Applying these mathematical expressions of availability requires capturing the input data over an extended period of use in order to obtain a statistically valid sample size. Since certification and accreditation take place over relatively short periods, an alternative is needed that presents sufficient evidence to substantiate a risk assessment from a security perspective.

Maintainability addresses the ability of an item to be retained in, or restored to, a specified condition when maintenance is performed by personnel having specified skill levels using prescribed procedures. There are three aspects to consider when measuring maintainability:

- The average corrective maintenance time required to restore the system to its mission-capable condition after a mission-critical failure.
- The average maintenance time required to restore the system after any failure that requires corrective maintenance. The average time to restore, considering all corrective maintenance, may be longer or shorter than the time for mission-critical failures.
- The manpower required to perform to perform maintenance, both corrective and preventive.

Another aspect to consider is the concept of workarounds, e.g., a change to procedure that avoids the anomaly, crash, or error. If the workaround allows the user to accomplish everything required, then the problem is basically cured, and the remaining time to diagnose and fix the problem should not be counted against measures such as availability and maintenance time. However, if the workaround involves a loss of function, it may be counted. Maintainability can be defined in terms of several different mathematical expressions, all of which depend upon long duration exposure to use (Table 3).

TYPE	PURPOSE	MATHEMATICAL EXPRESSION
Mean Repair Time (MRT)	Addresses all maintenance actions needed to correct a malfunction, including preparing for test, troubleshooting, removing and replacing components, repairing, adjusting, re-assembly, alignment, adjustment, and checkout. MRT does not include maintenance, supply or administrative delays.	$MRT = \frac{\text{Corrective Repair Hours}}{\text{Corrective Maintenance Events}}$
Mean Restart Time – Software Intensive (MRT_{SI})	Used to address the average elapsed time required to restart a software-intensive system. May also be represented as cold start MRT (MRT_C) and warm start MRT (MRT_W).	$MRT_{SI} = \frac{\text{Total Elapsed Time to Restart System}}{\text{Total Number of Software Restarts}}$
Mean Time to Restore Function (MTTRF)	Represents the average time required, as the result of a critical failure, to restore a system to full operating status.	$MTTRF = \frac{\text{Total critical restore time}}{\text{Number of critical failures}}$
Mean Corrective Maintenance Time For Operational Mission Faults-Software ($MCMTOMF_{SW}$)	Represents the average elapsed time needed to restore a software-intensive system following an operational mission software fault. This may include the time to restore all processes, functions, files, and databases to a tactically useful state as well as the time required to physically reboot the system following an operational mission software fault.	$MCMTOMFsw = \frac{\text{Total Elapsed Time to Restore Software-Intensive Systems After a Software OMF}}{\text{Total Number of Software OMFs}}$

TABLE 3. Mathematical Expressions of Maintainability.

MTTRF calculations include administrative and logistics delay times in addition to all maintenance actions needed to correct a malfunction, including testing to reproduce the error, diagnosis and troubleshooting, removing and replacing components, switching software levels (including forward and backward), rebooting, reorganizing, repairing, adjusting, re-assembly, alignment, and checkout/testing of the newly-repaired system. As will availability, some critical failures may affect only a portion of functionality or users. In those cases, it would be permissible to weight the MTTRF calculation accordingly.