Top ~~Ten~~ Eleven Ways to Manage Technical Risk

NAVSO P-3686
October 1998

Office of the Assistant Secretary
of the Navy (RD&A)
Acquisition and Business Management

Table of Contents

Acknowledgments

Project Sponsor

Elliott B. Branch, Office of the Assistant Secretary of the Navy (Research, Development & Acquisition) Acquisition and Business Management, Executive Director

Editor

Douglas O. Patterson, Office of the Assistant Secretary of the Navy (Research, Development & Acquisition) Acquisition and Business Management, Technical Director

Technical Writers

William A. Finn, EG&G Services
Edward I. Smith, EG&G Services
Toshio Oishi, IIT Research Institute
Eric Grothues, Naval Warfare Assessment Station

Contributors

CAPT Larrie Cable, USN, PEO(A) PMA-299
Frank Doherty, SPAWAR PMW-163
James Collis, NAVAIR 1.3.2
George Clessas, PEO(T) PMA-201
Barbara Smith PEO(A) PMA-275
John McGarry, Naval Undersea Warfare Center
Mike Wheeler, Naval Warfare Assessment Station
Lou Simpleman, Institute for Defense Analysis
Maye A. Hardin, By Design
Maria Zabko, EG&G Services

Contributing Navy Offices

PEO(A) PMA-271
PEO(A) PMA-275
PEO(A) PMA-276
PEO(A) PMA-290
PEO(A) PMA-299
PEO(CU) PMA-280
PEO(DD-21) PMS-500
PEO(MIW) PMS-407
PEO(SUB) PMS-401
PEO(SUB) PMS-411
PEO(SUB) PMS-450
PEO(T) PMA-201
PEO(T) PMA-259
PEO(T) PMA-265
PEO(T) PMA-272
PEO(TAD)-D
PEO(TAD) PMS-422
PEO(USW) PMS-403
PEO(USW) PMS-404
PEO(USW) PMS-415
SPAWAR PMW-163
SPAWAR PMW-183
DRPM AAAV
DRPM AEGIS PMS-400

Introduction

In recent years, risk management has been increasingly emphasized by the DoD as a critical tool for assuring program success. Whereas "risk management" is a general term encompassing all the different areas of risk management, this document focuses specifically on the single aspect of Technical Risk Management. Although managing risk for all aspects of a program is critical, technical risk is perhaps the most important area of risk management because technical risk, and the degree to which technical processes can be controlled, is a significant driver of all other program risks. Unfortunately, technical risk and the importance of controlling critical technical processes are generally not well understood within the DoD acquisition community, nor is adequate guidance on these considerations readily available.

In response to these shortcomings, and the need to improve the efficiency and effectiveness of the acquisition process, this publication offers a single source of concise explanations and clear descriptions of steps one can take to establish and implement core technical risk management functions. It contains baseline information, explanations, and best practices that contribute to a well founded technical risk management program – invaluable to program managers overwhelmed by the magnitude of information and guidance available on the broad subject of risk management today. In addition, as an aid to the reader, Appendix A contains the Risk Management Requirements from DoDD 5000.1 and DoD 5000.2-R, implemented by SECNAVINST 5000.2B.

Each chapter addresses specific technical risk areas. Although developed for Department of the Navy program managers and their staffs, this document should be equally useful to contractor program managers. The fundamentals contained herein are applicable to all acquisition efforts, both large and small.

Chapter 1

Choose an Approach

What is the Relationship Between "*Approach*" and Technical Risk?

The choice of an approach for managing program technical risk should be made as soon as possible. DoDD 5000.1 mandates that the Program Manager (PM) develop a risk management approach "before decision authorities can authorize a program to proceed into the next phase of the acquisition process." All aspects of a risk management program are, in turn, determined by the approach selected.

Delaying selection of a specific approach for managing technical risk will cause a program to flounder, especially if the contractor and the Government are following two different approaches. Further, the Integrated Product Team (IPT) cannot function successfully unless all members of the team – contractor and Government alike – are using a common approach.

Although the Defense Acquisition Deskbook offers the PM several approaches to risk management covering a broad spectrum of program risks, only three approaches have been selected for inclusion in this publication. Why? Results of a 1997 survey of risk management in 41 Department of the Navy (DoN) programs revealed that the following three approaches to managing program *technical* risk represent those used almost exclusively by DoN PMs.

3 primary approaches to technical risk management

1. **Critical Process**: Technical risk management conducted primarily by assessing contractor critical design, test, and production processes against industry best practices and metrics, with the degree of variance determining the level of risk. These critical processes are generally not tailored for individual Work Breakdown Structure (WBS) elements.

2. **Product (Work Breakdown Structure)**: Technical risk management based on individual product or WBS elements, with risk assessments based on deviations from a cost and schedule baseline. Risk is expressed as a "probability estimate" rather than as a degree of process variance from a best practice.

3. **Integrated Process/Product (WBS)**: Technical risk management based on

specific critical processes affecting individual WBS elements. These critical design, test, and production processes are assessed against industry best practices and metrics, with the degree of variance determining the level of risk. These approaches are described in the remainder of this chapter.

The Critical Process Approach

This approach is used to identify and analyze program technical risks by assessing the amount of variance between the contractor's design, test and production processes (i.e., those not related to individual WBS elements) and industry Best Practices. Success of any risk reduction efforts associated with this technique will depend on the contractor's ability and willingness to make a concerted effort to replace any deficient engineering practices and procedures with industry Best Practices. Chapter 5 contains a list of several fundamental engineering design, test and production Critical Processes with associated Best Practices and "Watch-Out-Fors." The Critical Processes were derived from a number of commercial and defense industry sources.

One of the primary benefits of this approach is that it addresses pervasive and subtle sources of risk in most DoD acquisition programs and uses fundamental engineering principles and proven procedures to reduce technical risks. Figure 1-1 illustrates a sample approach.

… utilizes proven engineering fundamentals

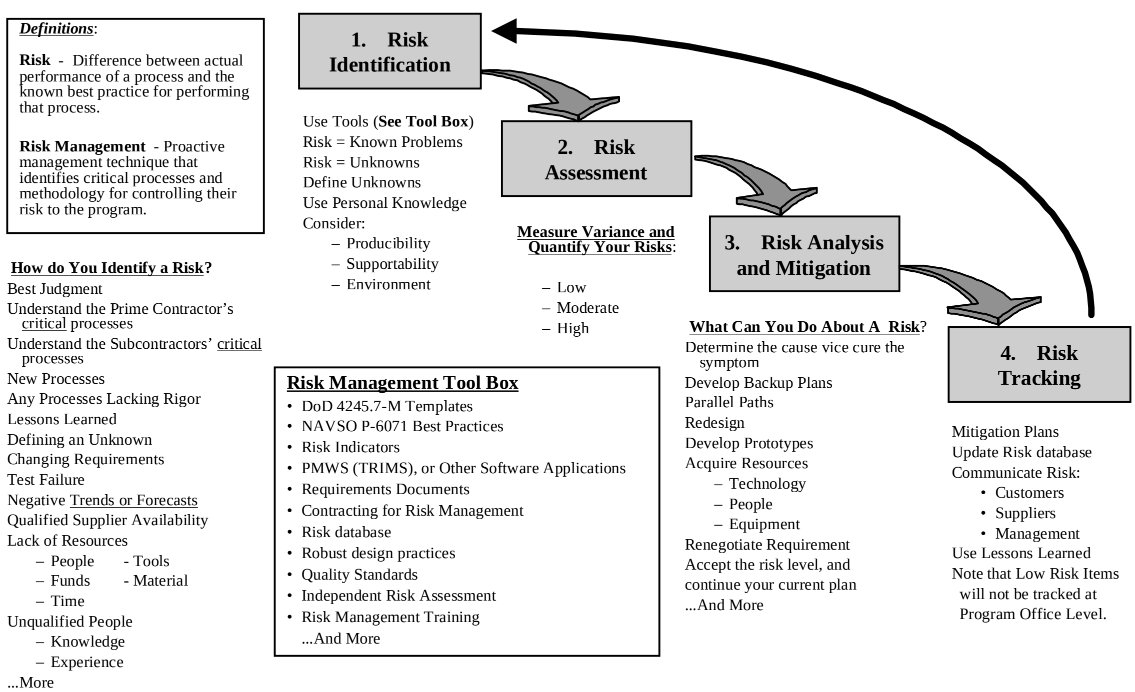


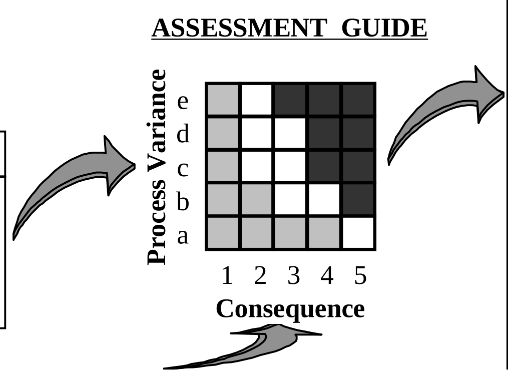
Figure 1-1. Critical Process Risk Management

Process Metrics, Best Practices and "Watch-Out-Fors" are used in conjunction with contract requirements and performance specifications to identify those technical processes that are critical to the program, and to establish a program baseline of contractor processes. This baseline should be developed using the fundamental engineering Critical Processes provided in Chapter 5 as a starting point and by reviewing and compiling additional Critical Processes in use by companies in both the defense and non-defense sectors.

The program baseline being used by the contractor should be determined by evaluating actual contractor performance, as opposed to stated policy. This program baseline should then be compared to a baseline of those industry-wide processes and practices that are critical to the program. The variances between the two baselines are indications of the technical process risk present in the program. These results should be documented in a standard format, such as a program-specific Risk Assessment Form (see Chapter 8), to facilitate the development of a risk handling/mitigation and risk tracking plan. Figure 1-2 illustrates a sample approach.

***CRITICAL PROCESS VARIANCE*:**

Level: What Is The Critical Process Variance from Known Standard?	
a	Minimal
b	Small
c	Acceptable
d	Large
e	Significant



RISK ASSESSMENT

HIGH - Major disruption likely. Different approach may be required. **Priority management attention required.**

MODERATE - Some disruption. Different approach may be required. Additional management attention may be needed.

LOW - Minimum impact. Minimum oversight needed to ensure risk remains low.

***CONSEQUENCE*:**
Given The Risk is Realized, What is the Magnitude of the Impact?

Level	Technical Performance	and/or	Schedule	and/or	Cost	and/or	Impact on Other Teams
1	Minimal or No Impact		Minimal or No Impact		Minimal or No Impact		None
2	Small with Some Reduction in Margin		Additional Resources Required; Able to Meet Need Dates		< 5%		Some Impact
3	Acceptable with Significant Reduction in Margin		Minor Slip in Key Milestone; Not Able to Meet Need Dates		5 - 7%		Moderate Impact
4	Large, No Remaining Margin		Major Slip in Key Milestone or Critical Path Impacted		> 7 - 10%		Major Impact
5	Significant		Can't Achieve Key Team or Major Program Milestone		> 10%		Significant Impact

Figure 1-2. Critical Process Risk Assessment

Final thought on the Critical Process approach

In summary, the critical process approach has many benefits; however, the critical processes normally are not directly related to the individual WBS product elements comprising the weapon system being developed and produced.

The Product (Work Breakdown Structure) Approach

DoD 5000.2-R requires that DoD programs tailor a program Work Breakdown Structure (WBS) for each program using the guidance in MIL-HDBK-881, "Work Breakdown Structure," of 2 January 1998. MIL-HDBK-881 defines the WBS as:

> *"…a product oriented family tree composed of hardware, software, services, data and facilities which results from systems engineering efforts during the acquisition of a defense material item. A WBS displays and defines the product(s) to be developed and/or produced and relates the [WBS] elements of work to be accomplished to each other and to the end product(s)."*

Functional processes are not WBS elements

A sound WBS clearly describes what the program manager wants to acquire. It has a logical structure and is tailored to a particular defense materiel item. As stated in MIL-HDBK-881, the WBS is product oriented. It addresses the products required, NOT the functional processes or costs associated with those products. For example, subjects, such as design engineering, requirements analysis, test engineering, etc. are not products. Rather, they are functional processes, each representing a discrete series of actions, with specific objectives, during product development and/or production. These are normally not identified as MIL-HDBK-881 WBS elements and as a result, generally do not receive adequate program consideration.

Section 2 of MIL-HDBK-881 states that the WBS provides a framework for specifying the technical objectives of the program by first defining the program in terms of hierarchically related, product oriented elements and the work processes required for their completion. Therefore, the emphasis on "product" is to define the product(s) to be developed and/or produced, and to relate the elements of work to be accomplished to each other and to the end product(s). Unfortunately, in this approach, programs frequently place little emphasis on "process."

WBS tends to be an after-the-fact measure of risk

A typical WBS technical risk management approach is based on the WBS products. Risk assessments and mitigation activities are conducted primarily on the individual WBS elements, with an emphasis on technology, product maturity or perceived quality, with little emphasis on related processes. Risk is typically expressed as a probability estimate rather than as a degree of process variance from a best practice. In the WBS approach, technical risks are identified, assessed, and tracked for individual WBS elements identified at their respective levels, primarily for impact on cost and schedule, and the resulting effect on the overall product. Since DoD programs are established around the WBS, the associated costs and schedule for each product can be readily baselined, against which risk can be measured as a deviation against cost and schedule performance. Taking the WBS to successively lower level entities will help to assure that all required products are identified in terms of cost and schedule performance (as well as operational performance) goals. In general, a typical WBS approach tends to be more reactive than proactive. Although a direct measurement of product performance against cost and schedule performance has its benefits, there are also some significant downsides to an approach in which processes are not considered. The WBS, by virtue of its inherent organizational properties, produces technical performance measurements that are, in essence, after-the-fact measures of risk. Also, by not focusing on processes, the overall risk to the program may not be identified until the program is in jeopardy.

As stated in DoD 5000.2-R, the WBS provides a framework for program and technical planning, cost estimating, resource allocations, performance measurements, and status reporting. Whereas the WBS is a good tool for measuring technical performance against cost and schedule, it is an incomplete measure of technical risk without considering processes. It is important to recognize that the WBS is a product of the systems engineering process, which emphasizes both product and process solutions required for the completion of technical objectives. However, history indicates that until recently, "process" solutions received too little emphasis.

The Integrated Process/Product Approach

The Integrated Process/Product approach to technical risk management is derived primarily from the Critical Process approach and incorporates some facets of the Product/WBS approach. The systems engineering function takes the lead in system development throughout any system's life cycle. The purpose of systems engineering is to define and design *process and product solutions* in terms of design, test and manufacturing requirements. The work breakdown structure provides a framework for specifying the technical objectives of the program by first defining the program in terms of hierarchically related, *product oriented elements* and the *work processes required* for their completion.

This emphasis on systems engineering, including processes and technical risk, along with process and product solutions, validates and supports the *importance of focusing on controlling the processes*, especially the prime contractor and subcontractors critical processes. Such a focus is necessary to encourage a proactive risk management program, one that acknowledges the importance of understanding and controlling the critical processes especially during the initial phases of product design and manufacture.

Integrates the best aspects of both approaches

In summary, the Critical Process Approach provides a proactive concentration on technical "drivers" and associated technical risks as measured by process variance. Integrating this approach into the Product Approach enables the critical processes to be directly related to the products comprising the weapon system being developed and produced. In this manner, the benefits of both approaches are realized. Product maturity is accelerated, technical risk is reduced, CAIV objectives are more easily met, schedule slippages are avoided, and the Program Manager reaches Milestone decision points with a higher level of confidence. See Table 1-1 for an overview of the advantages and disadvantages of all three approaches.

Table 1-1. Comparison of Approaches

Approach	Advantages	Disadvantages
Process	• Proactive focus on critical processes • Encourages market search for best practices/benchmarks • Reliance on fundamental design, test and manufacturing principles • Addresses pervasive and subtle sources of risk • Technical discipline will pay dividends in cost and schedule benefits	• Less emphasis on the product oriented elements of a program • Perception that technical issues dilute the importance of cost and schedule
Product (WBS)	• Commonly accepted approach using a logical, product oriented structure • Relates the elements of work to be accomplished to each other and to the end product • Separates a defense materiel item into its component parts • Allows tracking of product items down to any level of interest	• Does not typically emphasize critical design and manufacturing processes, or product cost • Risk is typically expressed as a probability estimate rather than a process variance • Delayed problem identification (reactive)
Integrated Process/ Product	• Maximizes the advantages of Process and Product approaches	• None significant

Chapter 2
Assign Accountability

What is the Relationship Between "*Accountability*" and Technical Risk?

In practice, most programs do not have an individual accountable to the Program Manager (PM) for risk management. More often than not, several team members may be assigned risk management responsibilities but do not have ownership or accountability in the risk management process. Therefore, it is imperative that a risk management focal point, accountable directly to the PM for the risk management program, be established and specifically identified in the program structure. Otherwise, risk management will quickly disintegrate and become an "Oh, by the way!" task until program risks have turned into program problems.

Risk Management Organization

The risk management team is not an organization separate from the program office. Rather, it is integrated with the program office and includes program office, prime contractor, field activity, and support contractor personnel operating toward a shared goal. A conceptual risk management organization, which shows relationships among members of the program risk management team, is provided in Figure 2-1.

Assign a Risk Management Coordinator

The key to establishing an effective risk organization is to formally assign and empower an individual whose primary role is managing risk. This individual, referred to as the Risk Management Coordinator, should be a higher-level program office person, such as the Deputy Program Manager (DPM), and should be accountable directly to the PM for all aspects of the risk program. The Risk Management Coordinator must have a level of authority which provides direct, unencumbered access to the PM and can cross organizational lines. The Risk Management Coordinator:

- Is the official point of contact and coordinator for the risk program
- Is responsible for reporting risk
- Is a subject matter expert on risk
- Maintains the risk management plan
- Coordinates risk training
- Does not need to be assigned full time

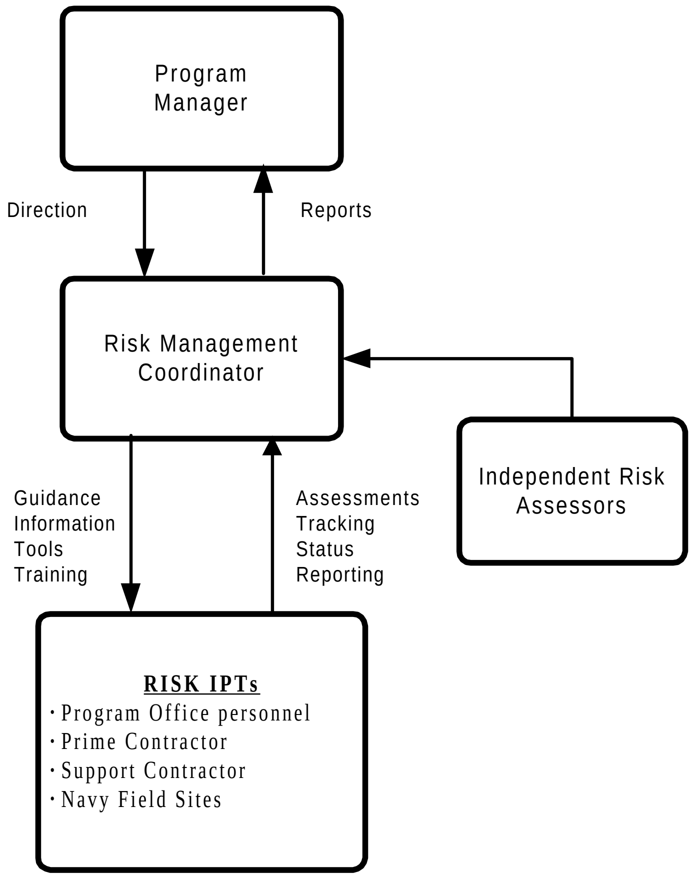


Figure 2-1. Conceptual Risk Management Organization

Risk Integrated Process Teams

Assign the correct people to IPTs – Not just bodies

Providing information to the Risk Management Coordinator are the actual IPTs responsible for implementing the risk program. These are comprised of experienced individuals from the different disciplines and functional areas within the program. Whereas these teams (or individuals) provide risk status and mitigation information to the Risk Management Coordinator, they are empowered to make recommendations and decisions regarding risk management activities, while reporting risk without the fear of reprisal. IPTs are responsible for:

- Providing to the Risk Management Coordinator the results of risk assessments and mitigation activities, using standard risk assessment forms
- Maintenance of the risk management database
- Implementing risk management practices and decisions, including those discussed at program and design reviews

Recommend a formal Risk Management Plan

It is imperative that everyone involved with the risk program understands his or her roles and responsibilities. Standard terminology, definitions, and formats are critical to the risk management process (see Chapter 8, "Assess, Mitigate, Report…"). The most effective method to do this is to document the risk process in a formal Risk Management Plan. Not only will a documented plan provide a standardized operating procedure for the risk effort, but it will provide continuity to the risk program as new personnel enter the program and others leave. The DoD Risk Management Working Group, chartered by the Office of the Under Secretary of Defense (Acquisition and Technology) (USD(A&T)) has developed a sample format for a Risk Management Plan. The sample plan is a compilation of several good risk management plans taken from DoD programs. The plan is designed to be tailored to fit individual program needs, and may be more comprehensive than many programs require. The DoD Risk Management Plan outline is available on-line in the DoD Deskbook. Additional information may be obtained from the DoD Risk Management Homepage at:

http://www.acq.osd.mil/te/programs/se/risk_management/index.htm

Chapter 3

Put Risk Management in the Contract

What is the Relationship Between "*The Contract*" and Technical Risk?

The elimination of many Military Specifications and Standards, the use of performance specifications and the shift of technical responsibility to contractors will not alone minimize program risk without explicit contractual requirements for risk management. The perception is that the transfer of responsibility to the contractor automatically reduces program risk. However, if a program fails because risk isn't managed well by the contractor, the Program Manager (PM) is ultimately responsible. The need for contractual requirements for risk management is recognized in both DoDD 5000.1 and DoD 5000.2-R.

The Request for Proposal

The Request for Proposal (RFP) should communicate to all Offerors the concept that risk management is an essential part of the Government's acquisition strategy. Before the draft RFP is developed, the PM should conduct a preliminary risk assessment to ensure that the program to be described in the RFP is executable within technical, schedule and budget constraints. Based on this assessment, the technical, schedule, and cost issues identified should be discussed at pre-proposal conference(s) before the draft RFP is released. In this way, critical risks inherent in the program can be identified and addressed in the RFP. In addition, this helps to establish key risk management contractual conditions as emphasized in the DoD 5000 series. During the pre-proposal conference, Offerors should be encouraged to identify all elements at any level that are expected to be moderate or high risk.

In the solicitation, PMs should ask Offerors to address:

- A risk management program
- An assessment of risks
- Risk mitigation plans, for moderate and high risks

In addition, the RFP should identify the requirement for periodic (define the frequency) risk assessment reports that would serve as inputs to the PM's risk assessment and monitoring processes, and ensure that risks are continuously assessed. Some programs require risk assessment reports for integration into quarterly Defense Acquisition Executive Summary reports.

Each RFP section is intended to elicit specific types of information from Offerors that will, when considered as a whole, permit selection of the best candidate to produce the goods or perform the services required by the Government. A number of sections of the RFP are key to risk management and are described as follows, including examples of typical clauses.

SECTION C - Description/Specifications/Statement of Work. This Section of the RFP includes any description or specifications needed. Statements describing risk management requirements may be included directly in Section C or by reference to the Statement of Work (SOW) or Statement of Objectives (SOO). A typical Section C clause is shown below:

> "The Offeror shall describe its proposed risk management program. The Offeror shall describe how they intend to identify, assess, mitigate, and monitor potential technical risks. Critical technical risks which may adversely impact cost, schedule, or performance shall be identified along with proposed risk mitigation methods for all risks identified as moderate or high."

Sample RFP Section C Clause

SECTION L - Instructions, Conditions, and Notices to Offerors. This Section of the RFP includes provisions, information and instructions to guide Offerors in preparing their proposals. Risk management requirements in Section L must be consistent with the rest of the RFP; such as tasking established in Section C, the SOW; evaluation criteria in Section M; and Special Provisions in Section H. The requirements must ensure the resulting proposals will form a solid basis for evaluation. The statements below provide examples from Navy programs for use in structuring Section L requirements to include risk management.

> **Volume I Part C - Management Proposal.**
> **Relevant Past/Present Performance:**
> "The Offeror shall demonstrate its past/present performance in critical requirements and processes and its ability to understand and resolve technical risk issues within its organizational structure, including interaction with its subcontractors. The Offeror shall discuss past/present performance in the implementation of risk reduction/mitigation efforts similar to those proposed for the reduction of all risks identified as moderate or high."

Sample RFP Section L Clauses

Volume I Part D - Technical Proposal.
"Risk Management. The Offeror shall provide a detailed description of the Risk management program to assure meeting the RFP requirements and objectives. The Offeror shall define and commit to the risk management program, including risk planning, identification, assessment, mitigation, and monitoring functions. The Offeror shall explain how its risk management process is related to the systems engineering and overall program management processes. The Offeror shall identify moderate and high technical risk areas, the known and potential risks in these areas, and the rationale for risk mitigation techniques proposed for these risk areas."

SECTION M - Evaluation Factors for Award. Section M notifies Offerors of the evaluation factors against which all proposals will be evaluated. These factors should be carefully structured to ensure that emphasis is placed on the critical factors described in Section L. They should set forth the relative importance of technical, cost (development versus production versus operational and support), schedule, management, and other factors, such as risk management and past performance, as set forth in the Source Selection Plan. The statement below provides an example for structuring Section M requirements to include risk management.

"The Government will evaluate the Offeror's proposed risk management program and plans for identifying, assessing, mitigating, and monitor risks, as well as proposed plans for mitigating those risks identified as moderate or high."

Sample RFP Section M Clause

When structuring Technical and Management evaluation criteria for Section M, risk management should be included as a factor or subfactor, as required by Part 3, paragraph 3.3.2 of DoD 5000.2-R.

Statement of Work/Statement of Objectives

The majority of existing Government contracts include a Statement of Work (SOW) that forms the basis for successful performance by the contractor and effective administration of the contract by the Government. A well-written SOW enhances the opportunity for all potential Offerors to compete equally for Government contracts and serves as the standard for determining if the contractor meets stated performance requirements.

Another concept called the Statement of Objectives (SOO) shifts the responsibility for preparing the SOW from the Government to the solicitation respondents. Recent DoD direction to lower Government costs encourages innovative contract options and flexible design solutions. The SOO captures the top level objectives of a solicitation - risk management, for instance - and allows the Offerors freedom in the structure and definition of SOW tasks as they apply to the proposed approach. The following paragraphs contain two SOW examples and one SOO example constructed from a number of Navy program solicitations.

SOW example

Risk Management and Reporting: The Contractor shall maintain a risk management program to assess risks associated with achievement of technical, cost, and schedule requirements. Specific risk management functions shall, at a minimum:

- Identify known and potential risks
- Assess risks, including a relative ranking by program impact and the establishment of critical thresholds
- Define methods or alternatives to mitigate or minimize these risks, including the identification of criteria upon which programmatic decisions can be based
- Track and report risk mitigation progress

The contractor's risk management program will be presented to the Government initially for concurrence and then in monthly updates and at in-process and other appropriate reviews.

SOW example

Risk Management: The Contractor shall implement a Risk Management Program in accordance with the XYZ Risk Management Plan using the Navy's *"Top Eleven Ways to ManageTechnical Risk"* publication as a guide. The initial set of Contractor-defined risks shall be updated as the Government or Contractor identifies new risks. The Contractor shall rank risks with respect to impact on performance, cost, and schedule and shall identify and develop mitigation plans for risk reduction/resolution.

SOO example

Risk Management Objectives:

- To develop and implement a risk management process with risk identification, assessment, mitigation and tracking/reporting functions
- To define and implement a risk assessment methodology that includes not only an understanding of cost, schedule and performance impacts but also a periodic reassessment of these impacts on identified risk areas
- To establish acceptable risk levels to be achieved
- To define risks and proposed risk mitigation steps for all items identified as moderate or high risk

Source Selection

DoD 5000.2-R states:

> *"Whenever applicable, risk reduction through the use of mature processes shall be a significant factor in source selection ---."*

The purpose of Source Selection is to select the contractor whose performance can be expected to meet the Government's requirements at an affordable price. The Source Selection process entails evaluating each Offeror's capability for meeting product and process technical, schedule, and cost requirements while identifying and managing inherent program risks.

The evaluation team must discriminate among Offerors based upon the risk associated with each Offeror's proposed approach for meeting Government requirements, including an evaluation of the Offeror's past and present performance record, to establish a level of confidence in the contractor's ability to perform the proposed effort. This evaluation should include consideration of:

- Product and process risk management approaches and associated risks determined by comparison with a Best Practices baseline
- Technical, cost and schedule assessments to estimate the additional resources (e.g., time, manpower loading, hardware, or special actions, such as additional analyses or tests, etc.) needed to control any risks that have medium or high risk ratings
- Past performance and recent improvements in the implementation of risk reduction/mitigation efforts similar to those being proposed for reducing risks identified as moderate or high for the program being proposed

Award Fee for Risk Management

Award fees, properly used, are a valuable tool for motivating contractors to improve performance while creating opportunities for improved Government – contractor communication, including ongoing feedback, thus permitting problems to be resolved sooner. Award fee discussions should be held on a regular basis; monthly or quarterly is usually recommended.

Guidelines for risk award fees

The award fee process can be successfully implemented on a range of contract goals and elements, including risk management. The guidelines below can help PMs establish a risk management program using award fee criteria:

- Analyze the SOW and attendant requirements to determine which contract performance requirements should be subject to awards
- Specify the criteria against which contractor performance will be measured
- From the total award fee amount to be made available, specify evaluation periods and the corresponding amount of award fee available each period
- Explain the general procedures that will be used to determine the earned award fee for each evaluation period

When analyzing the SOW and attendant requirements, an important first step is the identification of critical areas of program risk. Chapter 5 of this publication provides an initial set of critical technical process risk areas that can be used as a starting point in this effort. As a general rule, historically high-risk processes and processes involved with new technologies are usually good candidates for consideration as award fee elements.

Tailor the contract performance elements (i.e., areas of critical program risk) selected for award fees to key events, then assign them to appropriate award fee periods. The results become the basis of the request for information from potential bidders, as contained in the Instructions to Offerors, without having to ask for extraneous detail. A well thought out list of critical risk areas provides an excellent roadmap for the solicitation.

Award fee contracts based on contractor process improvements normally require some objective measurements to use as a basis for evaluation and award fee percentage determination. Give the contractor regular, structured feedback to preclude great disparity between what the contractor expects as an award fee payment and what the Government actually pays.

The simplicity of this approach is the very characteristic that makes the use of award fee criteria to establish a technical risk management program so effective.
Table 3-1 provides guidance for using award fee criteria in implementing technical risk.

Table 3-1. Award Fee Considerations

Best Practice

- Performance Feedback – Regular, structured feedback to prime contractors on their performance with respect to award fee criteria at significant program reviews
- Process Improvement – Process improvements can only be achieved if process changes are implemented
  - Verify implementation via test results documentation and operational use
  - Witness the actual implementation of new processes and procedures
- Award fee flowed down to subcontractors

Watch Out For

- No regular performance feedback provided by the Government to the prime contractor during the first evaluation period
- Award fee contracts based on contractor process improvements without objective measurements to use as a basis for evaluation and award fee determination
- Relatively short contract performance periods, making it difficult to establish a metric baseline, implement a process change and validate an actual improvement in the resulting metric during the contract period

Chapter 4
Mandate Training

What is the Relationship Between "*Training*" and Technical Risk?

It is often assumed that Government and contractor program staffs, as acquisition professionals, understand risk management. Given the nuances and complexities of risk management, most personnel perceive risk management differently due to varying backgrounds, experiences, and training. In order to integrate these variances, a formal indoctrination and/or awareness training in risk management is essential. All key Government and contractor personnel should understand their roles in the implementation of the risk management program, as well as the goals, strategies, roles, and responsibilities of the risk management team. Team members who are not "talking the same language" will result in a risk management effort that is poorly executed and ineffective.

Defense Acquisition University Courses

As in any organized effort, be it for a sports event or business venture, training is imperative for the success of the team or individuals involved. DoD risk management is no different – an inadequately trained staff is prone to failure. DoD has recognized the importance of training their acquisition professionals, and has established mandatory training standards in DoD 5000.52-M, "Acquisition Career Development Program," November 1995. The Defense Acquisition University (DAU), as established by the Under Secretary of Defense (Acquisition and Technology), provides a structured sequence of courses needed to meet the mandatory and desired training standards established in DoD 5000.52-M. These courses are designed to provide program office personnel with core and specialized knowledge in their functional areas, with higher level courses placing an emphasis on managing the acquisition process and learning the latest acquisition methods being implemented. Therefore, the program manager should ensure that, as minimum, key program office personnel involved with risk management attend these courses, which include:

- Production and Quality Management
- Logistics Fundamentals
- Fundamentals of System Acquisition Management
- Introduction to Acquisition Workforce Test and Evaluation

To enroll in the courses, program offices submit a Department of the Navy Acquisition Training Registration sheet (DACM1) to their command acquisition training representatives. Further information and course schedules can be obtained on the World Wide Web at:

http://dacm.secnav.navy.mil

Program Training

While training in the DoD Acquisition Professional Courses provides the big picture, it is also imperative that personnel be trained to their program's specific risk requirements and objectives. This is necessary to ensure that all personnel responsible for the implementation of risk management understand the program objectives, expectations, goals, terminology, formats, etc. regarding risk management. This training should not be limited to program office personnel, but includes the prime contractor, subcontractors, support contractors, and supporting field activities.

Training, as a minimum should provide instruction in the following:

- Background and introduction to risk management
- Program office risk organizational structure and responsibilities
- Concept and approach
- Awareness of latest techniques (through attendance at symposiums, seminars, workshops, etc.)
- Program definitions and terminology
- Risk assessment tools used by the program office
- Use of the risk management database. This should also include hands-on instruction on the use of the risk database and tracking system

The program office, prime contractor, or a support contractor can provide risk training; however, care should be taken in the selection of the training source. The training source(s) should be subject matter experts in risk management and be familiar with the program's operations.

As emphasized throughout this publication, all personnel responsible for planning and executing the risk management program must talk the same language, have an understanding of what the risk program's objectives are, and understand how to use the various tools required to identify, assess, mitigate and track risk. As in any venture, this training is critical to achieving the objectives for the successful execution of a risk management program.

Chapter 5

Practice Engineering Fundamentals

What is the Relationship Between *"Engineering Fundamentals"* and Technical Risk?

Engineering fundamentals are the basic disciplined design, test and production practices that have been proven through experience to be critical for risk avoidance. Experience has also shown that many of these fundamentals are not well understood by either the Government or industry. As a result, many program risks are derived from early management decisions regarding the application of these engineering fundamentals.

Critical Technical Processes

Critical processes are a continuum of interrelated and interdependent disciplines. A failure to perform well in one area may result in failure to do well in all areas. A high-risk program may result causing deployment of the product to be delayed, with degraded product performance, and at greater cost than planned. Risk is eliminated or reduced when the deficient industrial process is corrected, and that correction is often effected at a level of detail not normally visible to the Program Manager (PM).

This chapter contains fundamental technical processes and the associated Best Practices and "Watch-Out-Fors" which have great influence on technical risk. These practices, though by no means comprehensive, do focus on key technical risk areas. Use of proven best practices to achieve product success leads to a more organized approach to accomplish these activities and places more management significance on them.

Experienced engineers and PMs are aware that there are some requirements, conditions, materials, types of equipment or parts, and processes that almost invariably create potential or actual risk, identified herein as "Watch-Out-Fors." Knowing these areas of potential or actual risk gives a PM additional early insight for developing risk management or risk mitigation plans.

The Best Practices and "Watch-Out-Fors" associated with critical industrial technical processes should be used as a starting point in developing a baseline of program specific contractor processes. The best practices associated with these critical processes can also serve as benchmarks with which to compare your program's baseline processes and results achieved versus desired goals. The following examples of critical processes for the Design, Test, and Production phases of a product's development are presented in this chapter.

DESIGN	TEST	PRODUCTION
• Design Reference Mission Profile • Trade-Studies • Design Analyses • Parts & Materials Selection • Design for Testability • Built-In-Test • Design Reviews • Thermal Analysis • Design Release • Computer Aided Design/Computer Aided Manufacturing	• Design Limit Qualification Testing • Test, Analyze, and Fix	• Manufacturing Plan • Rapid Prototyping • Manufacturing Process Proofing/Qualification • Conformal Coating for Printed Wiring/Circuit Assemblies • Subcontractor Control • Tool Planning • Special Test Equipment • Manufacturing Screening • Failure Reporting, Analysis and Corrective Action

Design Reference Mission Profile

A Design Reference Mission Profile (DRMP) is a hypothetical profile consisting of time-phased functional and environmental profiles derived from multiple or variable missions and the total envelope of environments to which the system will be exposed. The DRMP becomes the basis for system and subsystem design and test requirements.

Best Practice

- Mission Profiles cover all system environments during its life cycle including operational, storage, handling, transportation, training, maintenance, and production
- Mission Profiles are defined in terms of time (duration and sequence), level of severity, and frequency of cycles
- Mission and System Profiles are detailed by the Government and contractor respectively, based on natural and induced environments (e.g., temperature, vibration, electromagnetic impulse, shock, and electrical transients)
- Profiles are the foundation for design and test requirements from system level to piece parts, including Commercial-Off-The-Shelf/Non-Developmental Items (COTS/NDIs)

Watch Out For

- DRMP environmental profiles that appear to be simply extracted from MIL-HDBK 810, "Environmental Test Methods and Engineering Guidelines," 31 July 1995
- Mission Profiles based on average natural environmental conditions rather than the more extreme conditions that may more accurately reflect operational requirements in the place/at the time of use, such as indicated by MIL-HDBK-310 "Global Climatic Data for Developing Military Products," 23 June 1997 and the National Climatic Data Center

Trade Studies

Trade Study are iterative series of studies performed to evaluate and validate concepts representing new technologies or processes, design alternatives, design simplification, ease of factory and field test, and compatibility with production processes. Trade studies culminate in a design that best balances need against what is realistically achievable and affordable.

Best Practice

- Trade studies are performed to evaluate alternatives and associated risks
- Trade studies consider producibility, supportability, reliability, cost and schedule as well as performance
- Trade studies are conducted using principles of modeling and simulation, experimental design and optimization theory
- Trade studies include sensitivity analyses of key performance and life cycle cost parameters
- Trade study alternatives are documented and formally included in design review documentation to ensure downstream traceability to design characteristics
- Trade studies are traceable to the DRMP and associated design requirements
- Quality Function Deployment techniques are used to identify key requirements when performing trade-offs

Watch Out For

- Use of new technologies without conducting trade-studies to identify risks
- Trade studies that do not include participation by appropriate engineering disciplines
- Product reliability, quality and supportability traded for cost, schedule and functional performance gains

Design Analyses

Design Analyses are performed to examine design parameters and their interaction with the environment. Included are risk-oriented analyses such as stress, worst case, thermal, structural, sneak circuit, and Failure Modes, Effects and Criticality Analysis (FMECA), which, if conducted properly, will ensure that reliable, low risk, mature designs are released.

Best Practice

- Validate new analysis modeling tools prior to use
- Conduct logic analysis on 100% of Integrated Circuits (ICs)
- Analyze 100% of IC outputs for ability to drive maximum expected load at rated speed and voltage levels
- ☞ Use Table 5-1 below to determine which design analyses should be performed

Watch Out For

- Analyses performed by inexperienced analysts
- Analyses performed using unproven software programs

Table 5-1. Objectives of Selected Design Analyses

Analyses	Objectives
• Reliability Prediction	• To evaluate alternative designs, assist in determining whether or not requirements can be achieved and for help in detecting over-stressed parts and/or critical areas
• Failure Modes, Effects and Criticality Analysis	• To identify design weaknesses by examining all failure modes using a bottom-up approach
• Worst Case Analysis	• To evaluate circuit tolerances based on simultaneous part variations
• Sneak Circuit Analysis	• To identify latent electrical circuit paths that cause wanted functions or inhibit wanted functions
• Fault Tree Analysis	• To identify effects of faults on system performance using a top-down approach
• Finite Element Analysis	• To assure material properties can withstand intended mechanical stresses in the intended environments
• Stress Analysis	• To determine or verify design integrity against conditional extremes or design behavior under various loads
• Thermal Stress Analysis (see Thermal Analysis)	• To determine or eliminate thermal overstress conditions; to verify compliance with derating criteria

Parts and Material Selection

The Parts and Material Selection utilizes a disciplined design process including adherence to firm derating criteria and the use of Qualified Manufacturers Lists (QML) to standardize parts selection.

Best Practice

- Use (QML) parts, particularly for applications requiring extended temperature ranges
- Electrical parameters of parts are characterized to requirements derived from the Design Reference Mission Profile to ensure that all selected parts are reliable for the proposed application (see Figure 6-3, Chapter 6)
- Derate all parts electrically and thermally
- A Preferred Parts List is established prior to detailed design
- Parts screening is tailored based on maturity
- Use highly integrated parts (e.g., Application Specific ICs (ASICs)) to reduce:
  - The number of individual discrete parts/chips
  - The number of interconnections
  - Size, power consumption, and cooling requirements, and
  - Failure rates
- Quality is measured by:
  - Certification by supplier
  - Compliance with EIA-623, “Procurement Quality of Solid State Components by Governments Contractors,” July 1994
  - Verification to historical data base
  - Particle Impact Noise Detection for cavity devices
  - Destructive Physical Analysis for construction analyses
- Strategy for parts obsolescence and technology insertion is established
- Vendor selection criteria established for non-QML parts considering:
  - Qualification, characterization and periodic testing data
  - Reliability/quality defect rates
  - Demonstrated process controls and continuous improvement program
  - Vendor production volume and history
- Minimum acceptable defects for in-coming electronic piece parts:
  - Maximum of 100 defective parts per million

Watch Out For

- Development of highly integrated parts unique to one specific acquisition development program
- Use of non-QML parts whenever QML parts are available
- Highly integrated parts that are not treated as a system of discrete parts to which the parts program requirements also apply
- Use of parts in environments not specified by the Original Equipment Manufacturer
- Variance in operating characteristics of commercial RF and analog parts
- Use of any parts near, at, or above their rated values, especially plastic encapsulated devices which reach higher junction temperatures than ceramic devices due to higher resistance to heat conduction
- Device frequency derating based on maximum overall operating temperature vs frequency rating which varies at different operating temperatures
- The use of parts beyond specified operating ranges by upscreening or uprating
- Designs using part technologies whose remaining life cycle will not support production and postproduction uses

Design for Testability

Designing for Testability assures that a product may be thoroughly tested with minimum effort, and that high confidence may be ascribed to test results. Testing ensures that a system has been properly manufactured and is ready for use, and that successful detection and isolation of a failure permits cost-effective repair.

Best Practice

- Perform testability analyses concurrently with design at all hardware and all maintenance levels
- Use Fault Tree Analysis, FMECA, and Dependency Modeling & Analysis to determine test point requirements and fault ambiguity group sizes
- Use standard maintenance busses to test equipment at all maintenance levels
- Use ASICs and other complex integrated circuits/chips with self-test capabilities
- Good testability design reflects the ability to:
  - Initialize the operating characteristics of a system by external means, e.g., disable an internal clock
  - Control internal functions of a system with external stimuli, e.g., break up feedback loops
  - Selectively access a system's internal partition and parts based on maintenance needs
- Evaluate Printed Wiring Board (PWB) testability using RAC publication Testability and Assessment Tool," 1991:
  - Converts scored and weighted rating of factors, including accessible vs. inaccessible nodes, proper documentation, complexity, removable vs. non-removable components, and different logic types (34 factors in all), to a possible total score of 100
  - The following testability scores illustrate this method

T-Scores for PWB Testability

Acceptable Score	**PWB Testability**
81 to 100	Very easy
66 to 80	Easy
Questionable Score	**PWB Testability**
46 to 65	Some difficulty
31 to 45	Average difficulty
Unacceptable Score	**PWB Testability**
11 to 30	Hard
1 to 10	Very hard
0 or less	Impossible to test/troubleshoot w/out cost penalties

Watch Out For

- Incompatibility between operational time constraints and time required to perform Built In Test (BIT)
- COTS/NDI testability design that is incompatible with mission needs and program life-cycle maintenance philosophy
- Testability design that results in special-purpose test equipment
- Circuit card assemblies and modules with test points that aren't accessible
- Circuit functions that don't fit on a single board
- Reverse funneling of tests
- Testability requirements for production are defined after design release

Built In Test

Built-In-Test (BIT) provides "built in" monitoring and fault isolation capabilities as integral features to the system design. BIT can be supplemented with embedded "expert system" technology that incorporates diagnostic logic/strategy into the prime system.

Best Practice

- BIT is compatible with other Automatic Test Equipment (ATE)
- Use BIT software:
  - For most flexible options (voting logic, sampling variations, filtering, etc.) to verify proper operation and identification of a failure or its cause
  - To minimize BIT hardware
  - To record BIT parameters
- Use multiplexing to simplify BIT circuitry
- Size the fault ambiguity group considering:
  - Mission requirements for reliability, repair time, down time, false alarm rate, etc.
  - Requirements for test equipment/manning at intermediate and depot maintenance levels
- Verify adequacy of the BIT circuit thresholds during development testing
- BIT should, as a minimum, provide:
  - 98% detection of all failures
  - Isolation to the lowest replaceable unit
  - Less than 0.1% false alarms
- Ratio of predicted to actual testability results 1:1
- Preliminary testability analysis - completed before PDR
- Detailed testability analysis - completed before CDR

Watch Out For

- High BIT effectiveness resulting in unacceptably high false alarm rates
- Inadequate time to perform BIT localization/diagnosis resulting in diminished BIT coverage and accuracy
- BIT design and analyses that fail to consider the effects of DRMP and worst-case variations of parameters, such as noise, part tolerance, and timing, especially as affected by age
- Inadequate BIT memory allocation
- Limitations to BIT coverage/effectiveness caused by:
  - Non-detectable parts (mechanical parts, redundant connector pins, decoupling capacitors, one-shot devices, etc.)
  - Power filtering circuits
  - Use of special test equipment (e.g., signal generators) to simulate operational input circuit conditions
  - Interface and/or compatibility problems between some equipment designs (e.g., digital vs analog)
- Unkeyed test connectors
- Test points without current limits
- Test points that are not protected against shorts to either adjacent test points or to ground
- Testing constraints that cause failures of one-shot devices, safety related circuits and physically restrained mechanical systems
- Methodology used to calculate BIT effectiveness

☞ (See Figure 5-1 for an illustration of this)

SYSTEM X

BIT	*Failures*	*Subsystem*
Yes	*100*	*A*
Yes	*80*	*B*
No	*15*	*C*
No	*5*	*D*
Total	*200*	*System X*

- *In this illustration, System X is designed for BIT detection of all failures of subsystems A and B, and none of the failures of subsystems C and D. The BIT effectiveness of System X can be calculated to be either 90% or 50% depending on the definition used.*
  - ◊ *90% BIT effectiveness is based on the percentage of the system total failures that are detectable. The total detectable failures of the BIT portions (of subsystems A and B) are 180 (i.e. 100+80) out of the system total of 200. 180/200= 90%.*
  - ◊ *50% BIT effectiveness is based on the percentage of the subsystems that can fail and be detected. Failure of only two subsystems (C & D) out of the four in the system are detectable by BIT. 2/4= 50%.*

Figure 5-1. BIT Design Based on Failure Rates

Design Reviews

A Design Review is a structured review process in which design analysis results, design margins and design maturity are evaluated to identify areas of risk, such as technology, design stresses, and producibility, prior to proceeding to the next phase of the development process.

Best Practice

- Formal procedures are established for Design Reviews
- Design Reviews are performed by independent and technically qualified personnel
- Entry and exit criteria are established
- Checklist and references are prepared
- Manufacturing, product assurance, logistics engineering, cost and other disciplines have equal authority to engineering in challenging design maturity
- Design Review requirements are flowed down to the subcontractors
- Subcontractors and customers participate in the design reviews
- Conduct design reviews as follows:
  - PDR: 20% of the design is complete
  - IDR: 50% of the design is complete
  - CDR: 95% of the design is complete

Watch Out For

- Reviews that are primarily programmatic in nature instead of technical
- Review schedules that are based on planned milestone dates
- Reviews held without review of analyses, assumptions, and processes
- Reviews held without review of trade-off studies, underlying data and risk assessments
- Reviews not formally documented and reported to management
- Reviews held by teams without adequate technical knowledge or representation of manufacturing, product assurance, supportability, etc.

Thermal Analysis

Thermal Analysis is one of the more critical analyses that is performed to eliminate thermal overstress conditions and to verify compliance with derating criteria. Thermal analyses are often supplemented with infrared scans, thermal paint, or the use of other measurement techniques to verify areas identified as critical.

Best Practice

- Determination and allocation of thermal loads and cooling requirements to lower-level equipment and parts are made based on the DRMP and the system self-generated heat
- Preliminary analyses are refined using actual power dissipation results as the thermal design matures
- The junction-to-case thermal resistance values of a device are used for the thermal analysis
- Thermal Survey (e.g., infrared scan) is conducted to verify the analysis

Watch Out For

- The use of device junction-to-ambient values for the thermal analysis, since this method is highly dependent on assumptions about coolant flow conditions
- A thermal analysis that does not take into account all modes (convection, conduction, radiation) and paths of heat transfer

Design Release

Design release is the point in the developmental stage of a product when creative design ceases and the product is released to production. Scheduling a design release is closely related to the status of other design activities such as design reviews, design for production, and configuration management.

Best Practice

- Design release process requires concurrent review by all technical disciplines
- Measurable key characteristics and parameters are identified on drawings, work instructions and process specifications
- Designs are released to production after:
  - Completion of all design reviews
  - Closeout of all corrective action items
  - Completion of all qualification testing
- A producible, supportable design is characterized by:
  - Stable design requirements
  - Completed assessment of design effects on current manufacturing processes, tooling and facilities
  - Completed producibility analysis
  - Completed rapid prototyping
- Completed analysis for compatibility with:
  - COTS/NDI interfaces
  - Subcontractor design interfaces
  - Form, Fit, and Function at all interfaces
- Design release practices, or equivalent, of the prime contractor are flowed down to the subcontractors

Watch Out For

- Design release based on manufacturing schedule
- Manufacturing drawings containing redlines
- Procurement for long lead items initiated with immature designs
- Drawings that are approved for release by engineering without review by all technical disciplines

Computer Aided Design/Computer Aided Manufacturing

Computer Aided Design/Computer Aided Manufacturing (CAD/CAM) introduces technical discipline throughout the design process to ensure success in complex development projects by integrating various design processes onto a common database. Included is the capability to perform special analyses, such as stress, vibration, thermal, noise, and weight, as well as to permit simulation modeling using finite element analysis and solids modeling. The outputs of this common database control manufacturing processes, tool design and design changes.

Best Practice

- Embed design rules in the CAD/CAM system
- Map CAD/CAM tools to the design and manufacturing processes
- Use compatible tools in an integrated CAD/CAM approach
- Use open architecture approach for software programs and data files
- Use new machine tools capable of being networked or upgraded to a network
- As a basis for procurement of new or upgraded CAD/CAM systems, sensitivity analyses are performed for various future scenarios (e.g., mainframe based versus Unix workstation-based, or NT based versus future cost to maintain and interconnect, 64 bit versus 32 bit math, links to ERP systems, etc.)
- 80% of design activity is computer based
- 100% of CAD drawings are CAM compatible
- Use common data exchange standards for 75% of processes
- All new machines networkable for CAM

Watch Out For

- CAD/CAM tools that operate in a stand-alone manner
- Failure to include total factory requirements and planned use for the CAD/CAM database
- Lack of a long-term growth plan to keep from being backed into a technological dead-end
- Proprietary Computer Numerically Controlled and Direct Numerically Controlled platforms and software architectures
- CAD/CAM systems which are non-standard for your industry, customers and suppliers
- Companies who will not be in business in several years

Design Limit Qualification Testing

Design Limit Qualification Testing is designed to ensure that system or subsystem designs meet performance requirements when exposed to environmental conditions expected at the extremes of the operating envelope, the "worst case" environments of the DRMP.

Best Practice

- Design limit/margin testing based on the DRMP, is integrated into the overall test plan, especially with engineering, reliability growth and life testing
- Design limit qualification tests are performed to ensure worst case specification requirements are met
- Highly Accelerated Life Tests (HALT) are performed to determine the design margins:
  - When operating at the expected worst case environments and usage conditions
  - To identify areas for corrective action
- Increased stress to failure conditions are included toward the end of Test, Analyze, and Fix (TAAF) testing to identify design margins
- Engineering development tests are performed beyond the design limits to measure the variance of the functional performance parameters under environmental extremes
- The failure mechanism of each failure, including stresses at the worst case specification limits, is understood

Watch Out For

- Design limit qualification testing environmental limits that are based on MIL-STDs and do not consider the DRMP
- In-service use of design limit qualification test units and other units that are stressed to a level resulting in inadequate remaining life
- Incompatibility of the COTS/NDIs qualification tests to the requirement
- Accelerated testing conditions which introduce failure modes not expected in normal use

Test, Analyze, and Fix

The Test, Analyze and Fix (TAAF) process is an iterative, closed loop reliability growth methodology. TAAF is accomplished primarily during engineering and manufacturing development. The process includes testing, analyzing test failures to determine cause of failure, redesigning to remove the cause, implementing the new design, and retesting to verify that the failure cause has been removed.

Best Practice

- Use of Duane or AMSAA Growth Models for the TAAF process
- Test facilities are capable of simulating all environmental extremes
- TAAF process starts at the lowest level of development and continues incrementally to higher assembly levels through the system level
- TAAF units are representative of production units
- TAAF process is integrated into the systems engineering development and test program to optimize the use of all assets, tests, and analyses.
- TAAF environments are based on worst case DRMP extremes, and normally include, as a minimum, vibration, temperature, shock, power cycling, input voltage variance, and output load
- TAAF is augmented by Failure Reporting and Corrective Action System to improve selected systems with a continuing history of poor reliability/performance
- HALT is performed at all hardware assembly levels as a development tool, and used as an alternative to TAAF to quickly identify design weaknesses and areas for improvement
- The mechanism of each failure, including stresses above the specification limits, is understood
- TAAF test resources should include between 2 to 10 Units-Under-Test (UUT), based on cost and complexity trade-off
- Ratio of TAAF test time at vibration and temperature extremes to total test times: $0.8 \leq \text{Ratio} \leq 1.0$
- Total calendar time (allocated and actual) to complete TAAF testing is approximately twice the number of test hours
- Test Time for each TAAF UUT is within 50% of the average time

☞ (Utilize Tri-Service Technical Brief “Test, Analyze and Fix (TAAF) Implementation,” January 1989)

Watch Out For

- Development programs with TAAF or HALT planned at the system level only
- TAAF planned or conducted in lieu of developmental/ exploratory engineering tests
- TAAF testing conducted with a limited sample size and a limited number of test hours/cycles
- Use of Bayesian approaches to shorten TAAF test time and to estimate reliability when the a-priori data is questionable
- A tendency to focus on statistical measures associated with TAAF and HALT, rather than using test results to identify and correct design deficiencies
- TAAF UUT and test facilities that are not conditioned/ groomed (burn in, screened, etc.) prior to test as planned for normal production
- Infant mortality failures are included in growth measurements
- The use of TAAF as a trial and error approach to correct a poor design
- The use of HALT to predict reliability

Manufacturing Plan

The Manufacturing Plan describes all actions required to produce, test and deliver acceptable systems on schedule and at minimum cost. The materials, fabrication flow, time in process, tools, test equipment, plant facilities and personnel skills are described and integrated into a logical sequence and schedule of events.

Best Practice

- Identification, during design, of key product characteristics and associated manufacturing process parameters and controls to minimize process variations and failure modes
- FMECA of the manufacturing process during design for defect prevention
- Specified manufacturing process variability (e.g. Cpk) is within the design tolerances
- Variations of test and measuring equipment are accounted for when determining process capability
- Rapid prototyping for reduced cycle time from design to production (see Rapid Prototyping).
- Design For Manufacturing and Assembly to develop simplified designs
- Design for agile manufacturing to quickly adapt to changes in production rate, cost and schedule.
- Contingency planning for disruption of incoming parts, variations in manufacturing quantities, and changes in manufacturing capabilities
- Controlled drawing release system instituted (see Design Release)
- Process proofing/qualification (see Manufacturing Process Proofing/Qualification)
- Product/process changes that require qualification are defined
- Flowcharts of manufacturing processes at the end of EMD, validated at the start of LRIP
- Facilities, manpower, and machine loading for full rate production are validated during LRIP. Production readiness reviews performed on critical processes
- Subcontractor process capabilities integrated into the prime contractor's process capabilities
- Specific product tests and inspections replaced with Statistical Process Controls (SPC) on a demonstrated capable and stable process
- Closed loop discrepancy reporting and corrective action system, including customer and subcontractor discrepancies
- Post production support plan established and maintained for:
  - Repair capability
  - Obsolescence of tools, test equipment and technology
  - Loss of contractor expertise and vendor base, and
  - Time/cost to reestablish production line

Metrics Include:

- Measurable key characteristics and parameters are identified on drawings, work instructions and process specification
- SPCs (e.g., Cpk>1.33) are established for key characteristics
- Critical processes under control prior to production implementation

Watch Out For

- Total cost of "hidden factory" for non-conforming materials
- A deficient Materials Requirements Planning system
- Inadequate response planning for subcontractor design and production process changes
- Establishment of SPC for key processes without use of statistical techniques (e.g., Design of Experiments, Taguchi, QFD) or adequate run time to determine variability of the process when stable
- Operator self-checks without a process to verify integrity of the system
- Planning which permits production work-a-rounds and fails to emphasize scheduled production outputs.

Rapid Prototyping

Rapid Prototyping utilizes physical prototypes created from computer generated three-dimensional models to help verify design robustness as well as reduce engineering costs during production activities associated with faulty or difficult to manufacture designs. The use of these prototypes includes functional testing, producibility, dimensional inspection, assembly training, as well as tool pattern development.

Best Practice

- Rapid prototyping technology used in developing a product from concept to manufacturing
- Used to reduce design cycle time, iterate design changes, check fit and interfaces, calculate mass properties and identify design deficiencies
- Used in manufacturing producibility studies, proof of tooling and fixtures, training, and as a visualization aid in the design of the evolving product
- Virtual reality prototypes are analyzed using CAD tools and physical parts are fabricated from the CAD three dimensional drawings and data prior to production

Watch Out For

- Rapid prototyping without three dimensional CAD data for precise geometric representation
- Two-dimensional CAD surface model used in lieu of the more complete three dimensional solid model
- Rapid prototyping without a support structure to sustain the part in place while it is being generated

Manufacturing Process Proofing/Qualification

Manufacturing Process Proofing/Qualification ensures the adequacy of production planning, tool design, assembly methods, finishing processes and personnel training before the start of rate production. This is done in a time frame that allows for design and configuration changes to be introduced into the product baseline.

Best Practice

- Proofing simulates actual production environments and conditions
- "Proof of Manufacturing" models used to verify that processes and procedures are compatible with the design configuration
- First article tests and inspections included as part of process proofing
- Conforming hardware consistently produced within the cost and time constraints for the production phase
- Key processes are proofed to assure key characteristics are within design tolerances
- Process proofing must occur with:
  - A new supplier
  - The relocation of a production line
  - Restart of a line after a significant interruption of production
  - New or modified test stations, tools, fixtures, and products
  - Baseline and subsequent changes to the manufacturing processes
  - Special processes (non-testable/non inspectable)
  - Conversion of manual to automated line

Watch Out For

- Process proofing that does not include integration into higher assemblies to assure proper fit and function at the end item level
- Changes in subcontractor processes that occur without notifying the prime
- The use of SPC to qualify or validate the manufacturing process in lieu of first article tests and inspections
- The use of acceptance tests in lieu of process proofing or performance of first article tests and inspections
- Performance of first article tests and inspections only when contractually required
- Attempts to cite the warranty provisions rather than actually proofing the processes
- Overly ambitious schedule for qualification of new products/sources

Conformal Coating for Printed Wiring Boards

A conformal coating is a thin film applied to the surface of a Printed Wiring Board or other assembly which offers a degree of protection from hostile environments such as moisture, dust, corrosives, solvents and physical stress.

Best Practice

- Use trade studies to weigh the effects of conformal coating on long-term reliability, safety, and rework costs against potential savings in production and repair costs
- Conformal coating is used in environments where contaminants cannot be adequately controlled, including manufacturing or testing facilities
- Match the type of conformal coating to the configuration, maintenance concept and the use environment of what you want to coat
- Inspection techniques in place to verify uniformity and completeness of conformal coating coverage

☞ (See Table 5-2 for selected coating properties)

Watch Out For

- Conformal coating used to meet hermetic requirements, since conformal coating is not hermetic or waterproof
- Manufacturing and/or testing processes lacking a Failure Reporting and Corrective Action System and quality system to ensure that precautions against contaminants are effective, especially on assemblies without conformal coating
- The application of conformal coating to a non-coated assembly without first assessing the effects on circuit operating frequencies, mechanical stresses, thermal hot spots, etc. that may increase failure rates
- The use of assemblies without conformal coating that contain critical analog circuits and/or high-power circuits, possibly creating safety hazards
- The use of conformal coating that is not compatible with the repair philosophy
- The toxicity and environmental friendliness of conformal coating, including its by-products
- Inadequate surface preparation and condition prior to application of conformal coating
- Improper masking prior to conformal coating

Table 5-2. Conformal Coating Material Properties

Coating / Properties	AR (Acrylic Resin)	UR (Urethane Resin)	ER (Epoxy Resin)	SR (Silicone Resin)	XY (Paraxylyene)
Nominal Thickness, Mils	1-3	1-3	1-3	2-8	0.6-1.0
Performance Under Humidity	Good	Good	Good	Good	Good
Resistance to Solvents	Poor	Good	Very Good	Fair	Excellent
Reparability	Excellent	Good	Poor	Fair	Poor
Application Characteristics	Excellent	Good	Fair	Fair	Fair
Volatile Organic Compound Exempt	Some	Some	Some	Some	All
Max. Continuous Operating Temp.	125C	125C	125C	200C	125C
Conveyor Processing Capability	Excellent	Poor to Fair	Poor to Fair	Poor to Fair	No

Source: Circuits Assembly, “A Focus on Conformal Coating” by Carl Tautscher, May 1997

Subcontractor Control

Reliance on subcontracting has made effective management of subcontractors critical to program success. Subcontractor Control includes the use of Integrated Product Teams, formal and informal design reviews, vendor conferences and subcontractor rating system databases.

Best Practice

- Subcontractor/supplier rating system with incentives for improved quality, reduced cost and timely delivery
- Flowdown of performance specification or detail Technical Data Package, depending on the acquisition strategy
- Subcontractors integrated into Integrated Product Teams to participate in the development of DRMP requirements
- Waiver of source and receiving inspections for subcontractors meeting certification requirements, depending on the product's criticality
- Subcontractor controls critical sub-tier suppliers
- Subcontractor notifies prime of design and process changes affecting key characteristics
- Metrics include subcontractor demonstrated process controls (e.g., Cpk > 1.33) for key characteristics

Watch Out For

- Procurement of critical material from an unapproved source
- Supplier performance rating does not consider the increased cost for defects discovered later in the prime's manufacturing process or after acceptance by the customer
- Subcontractor performance rating based primarily on cost, schedule and receiving inspection (vice performance requirements)
- Subcontractor process capability not verified
- Subcontractor decertification process is delinquent

Tool Planning

Tool Planning encompasses those activities associated with establishing a detailed, comprehensive plan for the design, development, implementation, and proof of program tooling. Tool planning is an integral part of the development process.

Best Practice

- Tools designed with CAD concurrent with product design
- Tool tolerances are at least 10% more restrictive than the hardware tolerances
- Measurement systems repeatability and reproducibility studies performed to establish the variability allowed to meet the key characteristic tolerances
- Tools are proofed, calibrated, certified and controlled
- Hard tooling validated prior to the start of production
- Tools are maintained with the aid of production statistical control charts
- Production tools are procured if the hardware is to be second sourced
- Minimize special tools and fixtures
- Metrics include:
  - Process capability Cpk > 1.33 for normal processes
  - Process capability Cpk > 1.67 for mission critical processes or for safety

Watch Out For

- Soft tooling used in production
- Calibration of tooling not traceable to a National standard and/or reference
- Master tooling not controlled

Special Test Equipment

Special Test Equipment (STE) is a key element of the manufacturing process used to test a final product for performance after it has completed in-process tests and inspections, final assembly and final visual inspection.

Best Practice

- STE is minimized
- ATE is developed for complex UUT, and considers test time limitations and accuracy
- STE accuracy/calibration must be traceable to known National measurement standard and/or references
- STE and applicable software are qualified, certified and controlled
- STE maintainability and maintenance concept defined concurrent with product design
- Life cycle functional and environmental profiles considered in STE design
- Design best practices are considered for critical STE
- Production demands are factored into STE design for reliability
- STE reliability target $\geq$ reliability of the system under test
- 4:1 minimum accuracy ratio between measurement levels (e.g., STE and UUT, standards and STE)

Watch Out For

- No fault repeatable loops
- STE software not validated
- STE production leads that impact increased rate production
- Root cause of STE discrepancies not understood
- STE false alarm rates
- STE not certified for acceptance testing
- Inadequate time between product CDR and STE delivery to support program schedule

Manufacturing Screening

Manufacturing Screening is a process for detecting in the factory, latent, intermittent, or incipient defects or flaws introduced by the manufacturing process. It normally involves the application of one or more accelerated environmental stresses designed to stimulate the product but within product design stress limits.

Best Practice

- Highly Accelerated Stress Screening (HASS) is performed as an environmental stress screen to precipitate and detect manufacturing defects
- HASS stress levels and profiles are determined from step stress HALT
- HASS precipitation screens are normally more severe than detection screens
- Product is operated and monitored during HASS
- The HASS screen effectiveness is proofed prior to production implementation
- HASS is performed with combined environment test equipment
- HASS stresses may be above design specification limits, but within the destruct limits, for example:
  - High rate thermal cycling
  - High level multi-axis vibration
  - Temperature dwells
  - Input power cycling at high voltage
  - Other margin stresses are considered when applicable to the product
- Alternative traditional environmental stress screening (ESS) guidelines for manufacturing defects may be in accordance with Tri-Service Technical Brief 002-93-08, "Environmental Stress Screening Guidelines," July 1993
- Parts Screening:
  - 100% screening required when defects exceed 100 PPM
  - 100% screening required when yields show lack of process control
  - Sample screening used when yields indicate a mature manufacturing process

Watch Out For

- Inadequate fatigue life remaining in the product after HASS
- HASS stresses that only simulate the field environment
- Environmental conditions that exceed the material properties of the product
- HASS that does not excite the low vibration frequencies

Failure Reporting, Analysis and Corrective Action

Failure Reporting, Analysis and Corrective Action is a closed loop process in which all failures of both hardware and software are formally reported. Analyses are performed to determine the root cause of the failure, and corrective actions are implemented and verified to prevent recurrence.

Best Practice

- Failure Reporting, Analysis, and Corrective Action System (FRACAS) implementation is consistent among the Government, prime contractor and subcontractors
- FRACAS is implemented from the part level through the system level throughout the system's life cycle
- Criticality of failures is prioritized in accordance with their individual impact on operational performance
- All failures are analyzed to sufficient depth to identify the underlying failure causes and necessary corrective actions
- Subcontractor failures and corrective actions are reported to the prime
- Prime contractor is involved in subcontractor closeout of critical failures
- Failure database accessible by customer, prime contractor and subcontractors
- Failure Review Board is composed of technical experts from each functional area
- Test requirements established for Retest-OK/Can-Not-Duplicate (RTOK/CND) failures

Metrics Include:

- 100% of failures undergo engineering analysis
- 100% of critical failures undergo laboratory analysis
- Failure analysis and proposed corrective action are completed:
  - $\leq$15 days for in-house analysis
  - $\leq$ 30 days for outsourced analysis
- Feedback from the field to the factory should be in $\leq$ 30 days

Watch Out For

- Deferring FRACAS to the production phase
- No time limit for failure analysis and closeout
- Verification of corrective action not part of failure closeout
- Failures classified as random are not analyzed
- Failure analysis required only when repetitive failures occur
- Pattern of RTOK/CND failures
- Exclusion of test equipment, GFE and COTS/NDI failures from FRACAS
- Engineering and lab analysis not considering:
  - History of previous failures
  - Related circuit part failures
  - Temperature and other environmental conditions at failure
  - Workmanship precipitated failures correctable by design changes
- RF and other high energy part failures often results from test setup difficulties
- Backlog of failures to be analyzed in the laboratory
- Failure Review Board (FRB) and Quality Review Board (QRB) not integrated to review effectiveness of both functional and non-functional failures
- Failure closeouts dependent on FRB/QRB decisions

Chapter 6
Understand COTS/NDI Applications

What is the Relationship Between "*COTS/NDI Applications*" and Technical Risk?

The use of Commercial Off-The-Shelf/Non-Developmental Items (COTS/NDIs) certainly has advantages, among them:

- Immediate availability of items, and
- Access to state-of-the-art technology available in the commercial sector, without incurring developmental costs

However, there are very clear risks associated with the use of COTS/NDIs. With continuously changing technology, traditional logistics support is often ineffective due to performance, configuration, and interface changes, coupled with a support system that takes too much time - often a period of time longer than the useful life of the item. Finally, since the use of COTS/NDIs is relatively new to the DoD, there is a paucity of data regarding the reliability, quality and performance of COTS/NDIs in a DoD environment. These risks can only be minimized through the knowledgeable and effective selection, integration, and qualification of COTS/NDIs.

Navy Experiences with COTS/NDI Applications

The information on the following pages represents lessons learned from Navy programs.

Design & Market Investigation

Best Practice

- Use Form, Fit, and Function requirements to query the market
- Begin market analysis early in program planning

Watch Out For

- Investigations slanted to make COTS the only acceptable choice
- COTS selections made without considering supportability and survivability
- Market investigations used for source selection/rejection

Selection

Best Practice

- Develop a procurement strategy for determining COTS viability for specific systems
- Be certain the strategy considers mission and environmental requirements

Watch Out For

- Determination of COTS suitability made in the absence of a standard selection process

Testing

Best Practice

- Inspect and test COTS/NDIs at incoming inspection
- Perform thorough testing through production
- Do not ship spares directly from the original vendor to the production / integration facility. Rather, spares should be functionally tested, preferably at the system level – and as a minimum at the subsystem level – using operational software. This will ensure design changes made by the vendor will not adversely affect the system during deployment

Watch Out For

- Standard test schedules, budgets, and documentation that fail to account for the additional testing needed for COTS/NDIs

Integration

Best Practice

- Require extensive compatibility testing of the product at both subassembly and system levels. What appears to be compatible at the subassembly level may not be so at the system level

Watch Out For

- The inherent difficulties in attempting a seamless integration of military items and COTS/NDI products

System Architecture

Best Practice

- Design systems to withstand the insertion of new technology
- Use an open-system architecture with strict adherence to COTS interface standards for hardware and software

Watch Out For

- Hardware/software systems designed with inadequate margins and too many "bells and whistles," making them prone to failure when new technology is introduced

Supportability

Best Practice

- Buy more spares than you think you need, because with COTS/NDIs, you will need them
- Communicate problems back to the vendors, (Many will take corrective action; their competitive position depends on it)
- Buy all spares during production and functionally test them at the system or subsystem level, using operational software
- Consider requiring vendor supplied drawings in enough detail to allow for an alternate source to protect against Diminishing Manufacturing Sources
- Define a COTS sparing policy for times when licenses and warranties expire before product spares are used

Watch Out For

- Supportability issues that plague COTS/NDI products due to frequent technology refresh cycles.
- The average life span of COTS items (which is between 6 and 24 months)
- COTS/NDIs which are not able to be repaired at lower levels of assembly (e.g., circuit card level); these assemblies are often obsolete in months
- Limited Sources for NDI spares
- Reliance on phone (800) technical support lines and limited training/maintenance documentation, especially in the field
- Replacement items that do not meet configuration requirements

COTS/NDI Product Maturity & Technology Refresh

Figure 6-1 provides a planning process flow chart that can be used to help determine when COTS technology should be refreshed or updated to maintain a supportable system.

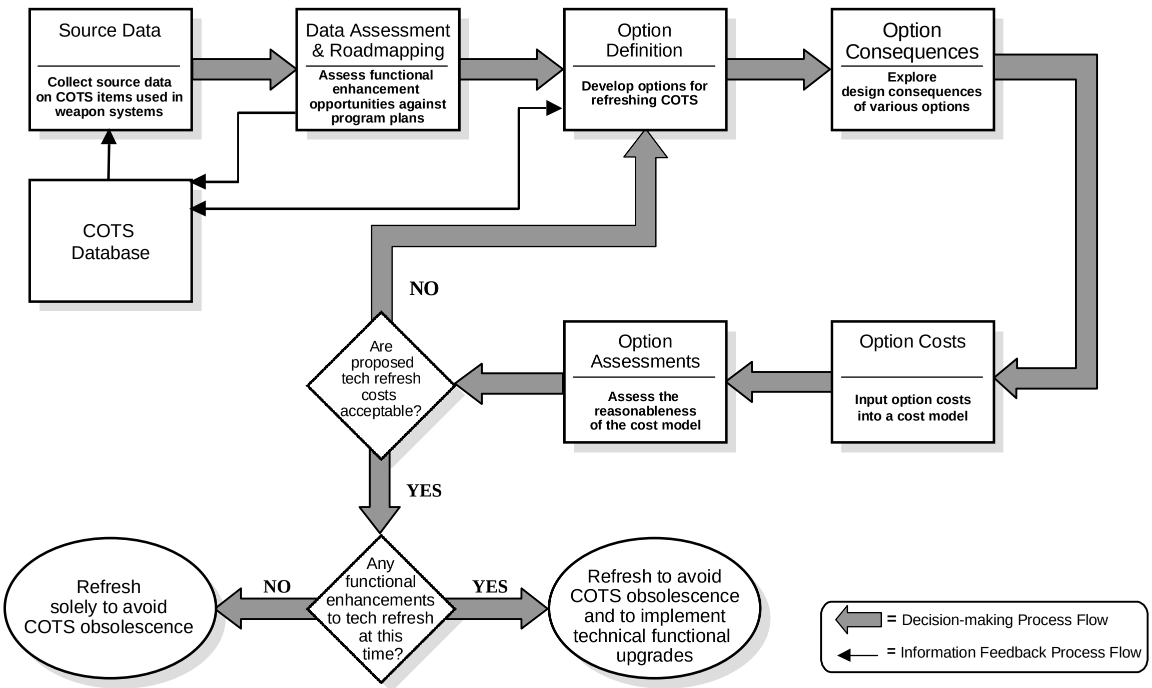


Figure 6-1. COTS Technology Refresh Planning Process

This process is intended to optimize the determination of when to refresh COTS items in order to keep program costs down and supportability high. Following this process allows the PM:

- To predict when COTS items/components may become unsupportable and require replacement
- To consider whether or not there are functional enhancements that could be realized by conducting a functional upgrade rather than simply refreshing existing COTS technology

Figure 6-2 provides metrics for determining the maturity of a COTS product. The four metrics, “state of the art”, “state of the practice”, “obsolete”, and “must refresh” indicate moderate risk, low risk, moderate risk and high risk scenarios respectively. The corresponding numbers in each metric correspond to blocks 1-10 below. Plotting a particular COTS product through the technical life cycle intervals will assist in determining if the COTS product should be refreshed or its design re-evaluated.

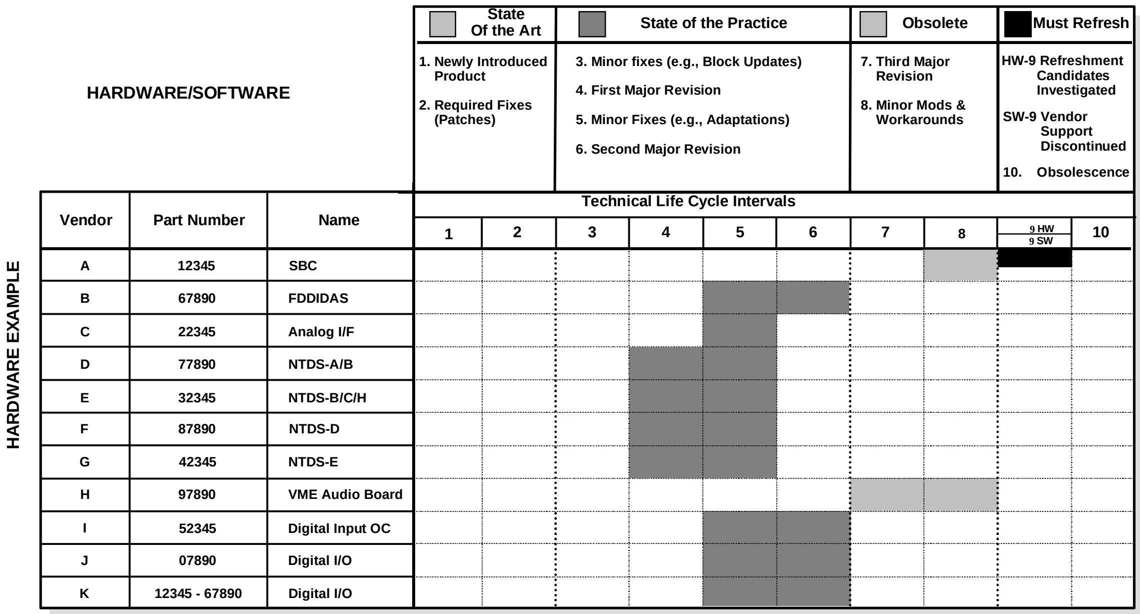


Figure 6-2. COTS Product Maturity Metric

Managing Risk Associated with Plastic Encapsulated Devices

The affordability, availability, and operational reliability of Plastic Encapsulated Devices (PEDs) influence their use in military designs. Today, PEDs may be as or more reliable than comparable Hermetically Sealed Microcircuit (HSM) ceramics when used under normal operating conditions. While PEDs are not necessarily COTS or NDIs, they are considered risk drivers and discussed in this section for the same reasons as COTS and NDIs; namely, their application, performance, and supportability in DoD applications are not fully known, and there is not enough data available today to prove PEDs will survive long-term dormant storage.

Know Your Vendor!

Choose parts from volume lines offered by reputable PEDs vendors — vendors recognized for using best commercial practices.

Some best practices associated with established PEDs manufacturers are:

- A demanding and like customer base
- Use of statistical process control techniques
- Conduct of qualification testing on their parts
- Available reliability, qualification, and process yield data
- Process/material change notifications supplied when process steps or materials are altered. A seemingly simple process change can cause significant quality variations from lot to lot. Typically, only high-volume customers receive this service, which usually excludes military customers

Part quality can vary significantly among vendors. Select PEDs from reputable vendors who meet the requirements of MIL-PRF performance specifications for QML production.

Always request and review vendor reliability and qualification test data. If vendor manufacturing data is not available or the vendor is not QML certified, use a discriminator test, such as the Highly Accelerated Stress Test (HAST) to assist in the comparison of PEDs quality/reliability and to select quality suppliers.

Conduct preconditioning accordance with EIA/JEDEC Standard No. A113-A (JESD22-A113-A), “Test Method A113-A, Preconditioning of Plastic Surface Mount Devices Prior to Reliability Testing,” June 1995; conduct HAST in accordance with EIA/JEDEC Standard No. 22-A110-A (JESD22-A110-A), Test Method A110-A, “Highly Accelerated Temperature and Humidity Stress Test,” April 1997. HAST quickly eliminates weak parts and accepts superior parts. It is the preferred test for simulating harsh military environments.

In addition to HAST, other commonly used commercial test methods are —

- High Temperature Operating Life (HTOL)
- High Temperature Storage (HTS) or Bake
- Solder Preconditioning
- Temperature Cycling (TC)
- Autoclave or Pressure Cooker
- Temperature Humidity Bias (THB)

Know Your Application!

- Select PEDs that can survive system life-cycle profiles and environments
- Ensure all environmental requirements are met. Provide environmental controls for long-term, unpowered conditions
- Prior to vendor selection, inform the vendor of the device’s intended mission life cycle applications, including storage and unpowered applications
- If a vendor’s existing qualification data does not adequately prove the reliability of the part in its intended environment, including storage, request additional

qualification/screening tests by the part vendor. It is more cost effective for the part vendor to conduct additional tests than to go to a third party
- Use the vendor (Original Equipment Manufacturer) to perform testing to benefit from the use of proprietary test methods, test equipment and knowledge of part design and construction

Emphasize the selection and use of quality molding compounds characterized by low stress, low cost, low contaminants, and low moisture absorption.
- The molding compound is a primary source of many problems, especially as it interfaces with the lead frame
- Vendors should use low stress/ionic contaminant epoxy compounds with strong adhesion to the lead frame
- Beware of a Coefficient of Thermal Expansion mismatch between lead frame and epoxy (encapsulant) and the epoxy and the die
- Vendors should report the glass transition temperature (Tg) and chemical properties of the epoxy (plastic)
- Choose a vendor who uses a high Tg epoxy

Choose a circuit card assembly process that is benign to the PED package.
- The surface mount technology soldering process can adversely affect the life of PEDs
- Minimize the amount of time the plastic body of the part is exposed to solder temperatures. If temperatures exceed the Tg of the plastic, the effects of the Coefficient of Thermal Expansion mismatch are magnified significantly
- Wave soldering or hand soldering (through-hole parts) is best; vapor phase, IR, convection reflow soldering (surface-mount) are worst. If surface-mount must be used, use conduction belt, hot bar, or hand soldering if possible
- Avoid mixed through-hole and surface mount boards where surface mount PEDs are sent through the fluxer and solder wave. This is a predominant source of ionic contamination and thermal stress
- ANSI/J-STD-020, “Classification of Moisture Sensitive Components,” provides recommended solder assembly reflow profiles to prevent “popcorning” and other related assembly damage
- External ionic contaminants picked up during the solder assembly process are the source of many problems

Be sure packaging requirements include dry bagging, vacuum bagging, and desiccants.
- As a minimum, keep PEDs sealed in a moisture-barrier bag with desiccant until attached to a circuit board. Consider keeping PEDs in a bag with desiccant at all levels of assembly until the completed assembly is loaded into the storage container and sealed with desiccant
- Determine the length of time parts/assemblies can be exposed to the ambient environment during manufacturing (*see* ANSI/J-STD-020). Some surface-mount plastic parts are fully saturated in as little as eight days, as is reflected by their moisture sensitivity classification (*see* ANSI/J-STD-020)

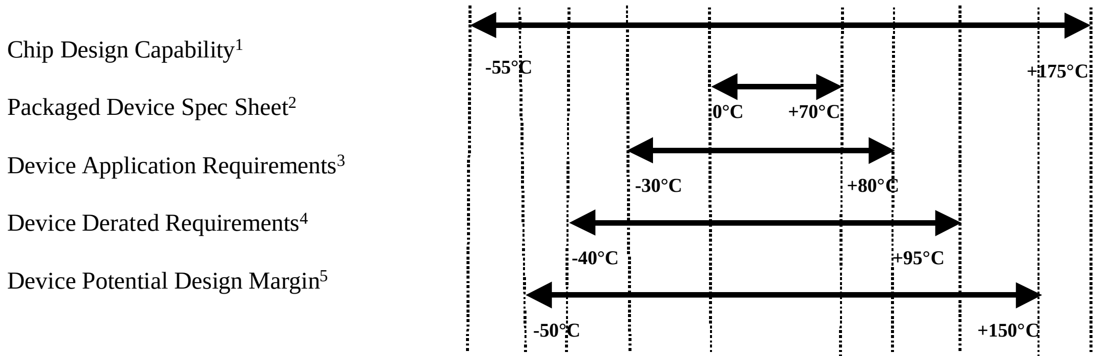
- Use ceramic packages rather than PEDs whenever cost is equivalent and an HSM is available. *But be aware* that the HSM may not be available for future buys, so if used, it should be interchangeable with a PED

Ensure PEDs are designed into the system from the beginning.

- It is very risky to arbitrarily replace a ceramic package with a PED
- The environmental parameters of the PED must be taken into account and necessary features must be added to the design to compensate for the more limited operating ranges of the PED
- Select and use open architecture and robust systems

And Don't Forget...

- PEDs should never be used outside their designed operating parameters. Upscreening a PED to a higher temperature is not recommended, and usually voids manufacturer warranties
- Manufacturing processes are planned to result in acceptable yields while maintaining warranty requirements. Higher operating temperatures can cause unacceptable variations in part operating characteristics and/or void warranty requirements
- Bond pads are usually the first area to fail during accelerated life testing
- There are <u>no</u> validated simulation models for making lifetime PED predictions
- Electrical parameters of parts are characterized to requirements derived from the Design Reference Mission Profile to ensure that all selected parts are reliable for the proposed application (see Figure 6-3)



Notes:
1. Chips are typically capable of and qualified for operation for these temperature ranges.
2. Chips packaged into devices are qualified according to their specification sheet for various temperature ranges, including commercial applications from 0° to 70°C.
3. Example of temperature ranges for the packaged device for an application in a specific system application.
4. Extended temperature ranges at which the device must be characterized/qualified for the specific system application for reliable derated capabilities.
5. Potential device design margin beyond specific system application, subject to manufacturing yield limitations.

Chip/device performance parameter typically widen with increasing temperatures. Proper circuit design requires proper derating, including trade-offs such as between temperature and operating frequency for reliable operations. Knowledge of the range in parameter values corresponding to the expected range in operating temperatures is essential. Otherwise, the risk of circuit instability and failure becomes unacceptable. Additionally, variations in the chip/device manufacturing processes require establishment of rated values with a safety margin below the maximum achievable.

Part manufacturers typically do not warranty their parts if used in applications more severe than the qualification levels at which the parts were characterized. Therefore, uprating or upscreening of a part is not recommended. The Qualified Manufacturer List (QML) program is designed to provide parts meeting the severe military environments.

Source: NAWC China Lake, Charles Barakat Brief, Parts Management and Encapsulated Devices, November 1996

Figure 6-3. Parts Rating and Characterization Process

Chapter 7

Establish Key Software Measures

What is the Relationship Between "*Software Measures*" and Technical Risk?

It is often believed that a risk free software program can be achieved through extensive test efforts. Whereas this approach may ultimately result in successful software development, it is generally a reactive risk approach. On the other hand, use of a software measurement process is a proactive approach for identifying risks before they become problems. This chapter contains a suggested approach for using key software measurement indicators as the foundation for identifying, assessing, mitigating and tracking software risks.

Measurement Selection for Tracking Risks

Experience shows that most project specific software risks can be grouped into categories that are basic or common to almost all projects. These common categories represent key concerns that must be managed on a day-to-day basis by the project manager. The six common software risk categories are listed in Table 7-1 along with examples for mapping these common risk categories to specific measure parameters. The measures are not intended to represent an exhaustive or required set of project management measures. However, they are measures that have repeatedly proven to be effective over a wide range of projects. In most cases, it is not practical to collect all of the possible measures for each risk category. Identification of the "best" set of measures for a project depends on a systematic evaluation of the potential measures with respect to the risks and relevant project characteristics. The measurement set cannot be predefined. Select the measure that best provides the desired insight based on both the required information and the project characteristics. See the end of this Chapter for a reference containing additional information.

Table 7-1. Risk Categories and Measures

ISSUE CATEGORY	MEASURE CATEGORY	MEASURE
Schedule and Progress Issues in this category relate to the completion of major milestones and individual work units. A project that falls behind schedule can usually only meet its original schedule by eliminating functionality or sacrificing quality.	Milestone Performance	Milestone Dates
	Work Unit Progress	Component Status Requirement Status Test Case Status Paths Tested Problem Report Status Reviews Completed Change Request Status
	Incremental Capability	Build content- Component Build content – Function
Resources and Cost Issues in this category relate to the balance between the work to be performed and personnel resources assigned to the project. A project that exceeds the budgeted effort usually can recover only by reducing functionality or sacrificing quality.	Personnel	Effort (Staff Months) Staff Experience Staff Turnover
	Financial Performance	Earned Value Cost
	Environment Availability (Tools & Facilities)	Resource Availability Dates Resource Utilization
Growth and Stability Issues in this category relate to the stability of the functionality or capability required of the software. It also relates to the volume of software delivered to provide the required capability. Stability includes changes in scope or quantity. An increase in software size usually requires increasing the applied resources or extending the project schedule.	Product Size and Stability	Lines of Code Components Words of Memory Database Size
	Functional Size and Stability	Requirements Function Points Change Request Workload

ISSUE CATEGORY	MEASURE CATEGORY	MEASURE
Product Quality Issues in this category relate to the ability of the delivered software product to support the user's needs without failure. Once a poor quality product is delivered, the burden of making it work usually falls on the sustaining engineering organization	Defects	Problem Reports Defect Density Failure Interval
	Complexity	Cyclomatic Complexity (Logic Paths)
Development Performance Issues in this category relate to the capability of the developer relative to project needs. A developer with a poor software development process or low productivity may have difficulty meeting aggressive schedule and cost objectives. More capable software developers are better able to deal with project changes.	Rework	Rework Size Rework Effort
	Process Maturity	Capability Maturity Model Level
	Productivity	Product Size/Effort Ratio Functional Size/Effort Ratio
Technical Adequacy Issues in this category relate to the viability of the proposed technical approach. It includes features such as software reuse, use of COTS software and components, and reliance on advanced software development processes. Cost increases and schedule delays may result if key elements of the proposed technical approach are not achieved.	Target Computer Resource Utilization	CPU Utilization CPU Throughput I/O Utilization I/O Throughput Memory Utilization Storage Utilization Response Time
	Technical Performance	Achieved Accuracy in Requirements (Concurrent Tasking, Data Handling, Signal Processing, etc.)
	Technology Impacts	Quantitative Impact of New Technology (NDI Utilization, Size by Origin, Cycle Time, etc.)

Software Measures

The following tables provide measurement descriptions for the measures listed in Table 7-1 and include:

- A definition of the measure
- Objectives to be achieved
- Specifications for the measure

Measure - Milestone Dates

The Milestone Dates measure consists of the start and end dates for software activities and events. The measure provides an easy to understand view of the status of scheduled software activities and events. Comparison of plan and actual milestone dates provides useful insight into both significant and repetitive schedule slips at the activity level.

Selection Guidance	Specification Guidance
Project Application • Basic measure applicable to all domains • Included in most DoD measurement policies and commercial measurement practices • Generally applicable to all sizes and types of projects • Useful during project planning, development, and sustaining engineering phases. Some sustaining engineering projects may be considered level of effort tasks and may not have associated milestones (or they may have only limited milestones such as date change assigned, date change closed) **Process Integration** • Required data is generally easily obtained from project scheduling systems and/or documentation. Data should be focused on software activities and events, particularly key items affecting the critical path or risk items • More detailed milestones provide a better indication of progress and allow earlier identification of problems • If dependency data is collected, slips in related activities can be more easily and accurately projected and assessed **Usually Applied During** • Requirements Analysis (Estimates and Actuals) • Design (Estimates and Actuals) • Implementation (Estimates and Actuals) • Integration and Test (Estimates and Actuals)	**Typical Data Items** • Start Date • End Date • Dependent Activity **Typical Attributes** • Version • Organization **Typical Aggregation Structure** • Software activity • Component **Typically Collected for Each** • Key software activity • Configuration Item (CI) or equivalent **Count Actuals Based On** • Customer sign-off • Action items closed • Documents baselined • Milestone review held • Successful completion of tasks

This Measure Answers Questions Such As:
- Is the current schedule realistic?
- How many activities are concurrently scheduled?
- How often has the schedule changed?
- What is the projected completion date for the project?

Measure - Component Status

The Component Status measure counts the number of software components that have completed a specific development activity. Early in the development activity, planning changes should be expected as the development activity is completed. Later in the process, an increase in the planned number of components can be an indication of unplanned or excessive growth. A comparison of planned and actual components is very effective for assessing development progress.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Used on medium to large projects • Useful during development and sustaining engineering phases • Tracking progress through early development activities, such as design or coding, is not generally done on projects without a design activity such as sustaining engineering projects that are focused on problem resolution or Commercial-Off-The-Shelf (COTS) integration projects • Tracking progress during the integration and test activities may be done for projects with major reuse or COTS integration **Process Integration** • Easier to collect if formal reviews, inspections, or walkthroughs are included in the development process • Data sometimes available from configuration management systems or development tools • Data is generally available if there is a mature and disciplined development process • Component status during test activities requires a disciplined testing process with separate tests per component(s) allocated to defined test sequences • Component status during test activities can be applied for each unique test sequence (i.e. CI test, integration test), including "dry-runs" • Component status during test activities is generally one of the more difficult work unit progress measures to collect since most integration and test activities are based on requirements or functions **Usually Applied During** • Requirements Analysis (Estimates) • Design Implementation (Estimates and Actuals) • Integration and Test (Estimates and Actuals)	**Typical Data Items** • Number of Units • Number of Units Complete **Typical Attributes** • Version • Software Activity **Typical Aggregation Structure** • Component **Typically Collected for Each** • CI or equivalent **Software Activity may be defined as** • Preliminary Design • Detailed Design • Code • Unit Test • CI Test **Count Actuals Based On** • Completion of component inspections, or walkthroughs • Successful completion of specified test • Release to configuration management • Resolution of action items

This Measure Answers Questions Such As:

- Are components completing development activities as scheduled?
- Is the planned rate of completion realistic?
- What components are behind schedule?

Measure - Requirement Status

The Requirement Status measure counts the number of defined requirements that have been allocated to software components and test cases and the number that have been successfully tested. The measure is an indication of software design and test progress. The measure addresses the degree to which required functionality has been successfully demonstrated against the specified requirements, as well as the amount of testing that has been performed. This measure provides an excellent measure of test progress. This measure is also known as "Breadth of Testing."

Selection Guidance

Project Application

- Applicable to all domains
- Useful during development and sustaining engineering phases. Not generally used on projects without a requirements or design activity such as sustaining engineering projects that are focused on problem resolution. Not generally used on projects in which requirements cannot be traced to test cases

Process Integration

- Requires disciplined requirements traceability and testing processes to implement successfully. Allocated requirements should be testable and mapped to test sequences. If an automated design tool is used, the data is more readily available
- Can be applied for each unique test sequence (i.e. CI test, integration test, system test, and regression test), including “dry-runs”
- One of the more difficult work unit progress measures to collect since requirements often do not directly map to components, test cases, and test procedures. It is also sometimes difficult to objectively determine if a requirement has been successfully tested
- Early in a project, the requirements baseline is limited to high-level specifications. Later on the requirements baseline expands and measurement data is traceable to components and test cases
- Some requirements may not be testable until late in the testing process. Others are not directly testable. Some may be verified by inspection
- Both stated and derived requirements may be counted

Usually Applied During

- Requirements Analysis (Estimates)
- Design, Integration and Test (Estimates and Actuals)
- Implementation (Estimates)

Specification Guidance

Typical Data Items

- Number of Requirements
- Number of Requirements Traced to Detailed Specifications
- Number of Requirements Traced to Software Components
- Number of Requirements Traced to Test Specifications
- Number of Requirements Tested Successfully

Typical Attributes

- Version
- Specification
- Test Sequence

Typical Aggregation Structure

- Function

Typically Collected for Each

- Requirement Specification

Count Actuals Based On

- Completion of specification review
- Baselining of specifications
- Baselining of Requirements Traceability Matrix
- Successful completion of all tests in the appropriate test sequence

This Measure Answers Questions Such As:

- Have all of the requirements been allocated to software components?
- Are the requirements being implemented and tested as scheduled?

Measure - Test Case Status

The Test Case Status measure counts the number of test cases that have been attempted and those that have been completed successfully. This measure can be used in conjunction with the Requirement Status measure to evaluate test progress. This measure allows assessment of software quality, based on the proportion of attempted test cases that are successfully executed. This measure is one of the best measures of test progress.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Generally applicable to all sizes and types of projects • Useful during development and sustaining engineering phases **Process Integration** • Need disciplined test planning and tracking processes to implement successfully • Can be applied for each unique test sequence (i.e. CI, test, integration test, system test, and regression test), including "dry-runs" • There should be a mapping between defined test cases and requirements. This allows an analysis of what functions are passing test and what ones are not • Easy to collect. Most projects define and allocate a quantifiable number of test cases to each software test sequence **Usually Applied During** • Implementation (Estimates and Actuals) • Integration and Test (Estimates and Actuals)	**Typical Data Items** • Number of Test Cases • Number of Test Cases Attempted • Number of Test Cases Passed **Typical Attributes** • Version • Test Sequence **Typical Aggregation Structure** • Software Activity (Test) **Typically Collected for Each** • Activity Test **Alternatives to Test Cases Include** • Test Procedures • Test Steps • Use/Case scenarios • Functional threads **Count Actuals Based On** • Successful completion of each test case in the appropriate test sequence

This Measure Answers Questions Such As:

- Is test progress sufficient to meet the schedule?
- Is the planned rate of testing realistic?
- What functions are behind schedule?

Measure - Paths Tested

The Paths Tested measure counts the number of logical paths successfully tested. The measure reports the degree to which the software has been successfully demonstrated and indicates the amount of testing that has been performed. This measure is also called "Depth of Testing."

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Applicable to most types of projects. Especially important for those with high reliability requirements, security implications, or catastrophic failure potential. • Not generally used for COTS or reused code • Useful during development and sustaining engineering phases **Process Integration** • Usually applied on a cumulative basis across all test sequences (i.e. CI test, integration test, system test, and regression test), so that each path is tested by the time all testing is complete • Often used in conjunction with Cyclomatic Complexity • Difficult to collect - requires the use of test tools that can verify test paths covered. These test tools often require instrumentation of the code • Difficult to use on large projects due to the large number of paths **Usually Applied During** • Implementation (Estimates and Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Number of Paths • Number of Paths Tested • Number of Paths Tested Successfully **Typical Attributes** • Version • Test Sequence **Typical Aggregation Structure** • Component **Typically Collected for Each** • Unit or equivalent **Alternative to Paths Include** • Executable Statements • Decisions **Count Actuals Based On** • Successful completion of each test in the appropriate test sequence

This Measure Answers Questions Such As:

- Have all of the paths been successfully tested?
- What percentages of the paths are represented in the testing approach?

Measure - Problem Report Status

The Problem Report Status measure counts the number of software problems reported and resolved. This measure provides an indication of product maturity and readiness for delivery. The rates at which problem reports are written and closed can be used to estimate test completion. This measure can also be used as an indication of the quality of the problem resolution process.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Applicable to all sizes and types of projects • Useful during development and sustaining engineering phases **Process Integration** • Many projects have acceptance criteria based on the number of open problem reports, by priority. This measure is useful in tracking to those requirements • The amount of test activity has a significant impact on this measure. Test personnel generally alternate between testing and fixing problems. You may want to normalize this measure using some measure of Test Progress • Data is generally available. Data is easier to collect when an automated problem tracking system is used • On development projects, data is generally available during integration and test. Problem report data is more difficult to collect earlier (during requirements analysis, design, and implementation), because the formal problem reporting system is usually not in place and rigidly enforced. When this data is available, it provides very good progress information. An inspection or peer review process can provide this information **Usually Applied During** • Requirements Analysis (Estimates and Actuals) • Design (Estimates and Actuals) • Implementation (Estimates and Actuals) • Integration and Test (Estimates and Actuals)	**Typical Data Items** • Number of Software Problems Reported • Number of Software Problems Resolved **Typical Attributes** • Version • Priority • Valid/lnvalid **Typical Aggregation Structure** • Component **Typically Collected for Each** • CI or equivalent **Count Actuals Based On** • Fix developed • Fix implemented • Fix integrated • Fix tested

This Measure Answers Questions Such As:
- Are known problem reports being closed at a sufficient rate to meet the test completion date?
- Is the product maturing (Is the problem report discovery rate going down)?
- When will testing be complete?
- What components have the most open problem reports?

Measure - Reviews Completed

The Reviews Completed measure counts the number of reviews successfully completed, including both internal developer and acquirer reviews. The measure provides an indication of progress in completing review activities.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Used on medium to large projects. Not generally used on projects integrating COTS and reusable software components • Useful during development and sustaining engineering phases **Process Integration** • Easy to collect if formal reviews are a part of the development process **Usually Applied During** • Requirements Analysis (Estimates and Actuals) • Design (Estimates and Actuals) • Implementation (Estimates and Actuals)	**Typical Data Items** • Number of Reviews • Number of Reviews Scheduled • Number of Reviews Completed Successfully **Typical Attributes** • Version **Typical Aggregation Structure** • Component • Software Activity **Typically Collected for Each** • CI or equivalent • Major activity **Alternatives to Reviews Include** • Inspections Walkthroughs **Count Actuals Based On** • Completion of review • Resolution of all associated action items

This Measure Answers Questions Such As:

- Are development review activities progressing as scheduled?
- Do the completed products meet the defined standards (Are components passing the reviews)?
- What components have failed their review?

Measure - Change Request Status

The Change Request Status measure counts the number of change requests, enhancements, or corrective action reports affecting a product. The measure provides an indication of the amount of rework required and performed. It only identifies the number of changes, and does not report on the functional impact of changes or the amount of effort required to implement them.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Applicable to all sizes of projects • Useful during the development phase. Often used for projects in the sustaining engineering phase. Not Resolved generally used for integration projects incorporating COTS and reused code **Process Integration** • Data should be available from most projects • Often used on iterative developments such as prototyping **Usually Applied During** • Requirements Analysis (Actuals) • Design (Actuals) • Implementation (Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Number of Software Change Requests written • Number of Software Change requests Resolved **Typical Attributes** • Version • Priority • Valid/lnvalid • Approved/Unapproved • Change Classification (defect correction, enhancement) **Typical Aggregation Structure** • Function **Typically Collected for Each** • Requirement Specification • Design Specification **Alternatives to Change Requests Include** • Enhancements • Corrective Action Reports **Count Actuals Based On** • Change implemented • Change integrated • Change tested

This Measure Answers Such Questions As:

- How many change requests have impacted the software?
- Are change requests being implemented at a sufficient rate to meet schedule?
- Is the trend of new change requests decreasing as the project nears completion?

Measure - Build Content - Component

The Build Content - Component measure identifies the components that are included in incremental builds. The measure indicates progress in the incremental products. Build content will often be deferred or removed in order to preserve the scheduled delivery date. It is easier to track incorporation of capability by component (rather than by function), since it is relatively easy to detect whether or not a component has been integrated. However, this provides less information, since the correlation between components and functionality is not always well defined.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Generally applicable to all sizes and types of projects • Useful during development and sustaining engineering phases **Process Integration** • Requires a formal, detailed list of content by increment. This content must be defined at the component level • Easy to collect, especially if the project has a detailed tracking mechanism • To effectively measure the content of the software at the version level, the lower level units that comprise the version must individually be complete with respect to defined criteria **Usually Applied During** • Design (Estimates) • Implementation (Estimates) • Integration and Test (Estimates and Actuals)	**Typical Data Items** • Number of Units • Number of Units Integrated successfully **Typical Attributes** • Version **Typical Aggregation Structure** • Component **Typically Collected for Each** • Unit or equivalent **Count Actuals Based On** • Successful integration • Successful testing

This Measure Answers Such Questions As:
- Are components being incorporated as scheduled?
- Will each increment contain the specified components?
- What components have to be deferred or eliminated?
- What components have been added?
- Is development risk being deferred?

Measure - Build Content- Function

The Build Content - Function measure identifies the functional content of incremental builds. The measure indicates the progress in the incorporation of incremental functionality. Build content will often be deferred or removed in order to preserve the scheduled delivery date.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Generally applicable to all sizes and types of projects • Useful during development and sustaining engineering phases **Process Integration** • Requires a formal, detailed list of functions by increment • Feasible to collect if the project has a detailed tracking mechanism. Easier to collect if use/case or functional threads are defined • It is often difficult to identify whether a function is incorporated in its entirety. A considerable amount of testing and analysis must be done to determine if all aspects of a function are incorporated **Usually Applied During** • Design (Estimates) • Implementation (Estimates) Integration and Test (Estimates and Actuals)	**Typical Data Items** • Number of Sub-Functions • Number of Sub-Functions Integrated Successfully **Typical Attributes** • Version **Typical Aggregation Structure** • Function **Typically Collected for Each** • Function or equivalent **Count Actuals Based On** • Successful integration • Successful testing

This Measure answers Questions Such As:
- Is functionality being incorporated as scheduled?
- Will each increment contain the specified functionality?
- What functionality has to be deferred?
- Is development risk being deferred?

Chapter 7

Resources and Cost

Measure- Effort

The Effort measure counts the number of hours or personnel applied to software tasks. This is a straightforward, generally understood measure. It can be categorized by activity as well as by product. This measure usually correlates directly with software cost, but can also be used to address other common issues including Schedule and Progress and Development Performance.

Selection Guidance	Specification Guidance
Project Application • Basic measure applicable to all domains • Included in most DoD measurement policies and commercial measurement practices • Generally applicable to all sizes and types of projects • Useful during project planning, development, and sustaining engineering phases. Some sustaining engineering projects with fixed staffing levels may not track this measure **Process Integration** • Data should be available from most projects at the system level • Data usually derived from a financial accounting and reporting system and/or separate time card system • All labor hours applied to the software tasks should be collected, including overtime. The overtime data is sometimes difficult to collect • Most effective when financial accounting reporting systems are directly tied to software products and activities at a low level of detail. Counting software personnel may be difficult because they may not be allocated to the project on a full-time basis or they may not be assigned to strictly software-related tasks • If labor hours are not explicitly provided, data may be approximated from staffing and/or cost data. Labor hours are sometimes considered proprietary data • The labor categories and activities that comprise the software tasks must be explicitly defined for each organization • Planning data is usually based on software estimation models or engineering judgment **Usually Applied During** • Requirements Analysis (Estimates and Actuals) • Design, Integration and Test (Estimates and Actuals) • Implementation (Estimates and Actuals)	**Typical Data Items** • Number of Labor Hours **Typical Attributes** • Organization • Labor Category **Typical Aggregation Structure** • Software Activity • Component **Typically Collected for Each** • Software Activity • CI or equivalent **Alternatives to Labor Hours Include** • Labor Days/Weeks/Months • Full Time Equivalents • Number of Personnel **Alternatives to WBS Elements include** • Software Activities **Count Actuals Based On** • End of financial reporting period

This Measure Answers Questions Such As:

- Are development resources being applied according to plan?
- Are certain tasks or activities taking more/less effort than expected, and is the effort profile realistic?

Measure - Staff Experience

The Staff Experience measure counts the total number of software personnel with experience in defined areas. The measure is used to determine whether sufficient experienced personnel are available and used. The experience factors are based on the requirements of each individual project (such as domain or language). Experience is usually measured in years, which does not always equate to capability.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Applicable to projects that require particular expertise to complete • Useful during project planning, development, and sustaining engineering phases **Process Integration** • Requires a personnel database that maintains experience data • Difficult to collect and keep up-to-date as people are added/removed from a project. Generally has to be • A matrix of project skill requirements versus current personnel skills may help to track this measure and identify necessary training areas **Usually Applied During** • Requirements Analysis (Actuals) • Design (Actuals) • Implementation (Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Number of Personnel • Number of Years of Experience **Typical Attributes** • Experience Factor **Typical Aggregation Structure** • Software Activity (Organization) **Typically Collected for Each** • Organization **Experience Factor May be Defined for** • Language • System Engineering • Domain • Hardware • Application • Platform • Length of Time Team Together **Count Actuals Based On** • Prior to contract award • During annual performance evaluation

This Measure Answers Questions Such As:

- Are sufficient experienced personnel available?
- Will additional training be required?

Measure - Staff Turnover

The Staff Turnover measure counts staff losses and gains. A large amount of turnover impacts learning curves, productivity, and the ability of the software developer to build the system with the resources provided within cost and schedule. This measure is most effective when used in conjunction with the Staff Experience measure. Losses of more experienced personnel are more critical.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Applicable to projects of all sizes and types • Useful during development and sustaining engineering phases **Process Integration** • Very difficult to collect on contractual projects – most developers consider this proprietary information. May be more readily available on in-house projects • It is useful to categorize the number of personnel lost into planned and unplanned losses, since most projects plan to add and remove personnel at various stages of the project **Usually Applied During** • Requirements Analysis (Actuals) • Design (Actuals) • Implementation (Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Number of Personnel • Number of Personnel Gained (per period) • Number of Personnel Lost (per period) **Typical Attributes** • None **Typical Aggregation Structure** • Software Activity (Organization) **Typically Collected for Each** • Organization **Count Actuals Based On** • End of financial reporting period • Organization restructuring or new organizational charts • End of project activities or milestones

This Measure Answers Questions Such As:

- How many people have been added/have left the project?
- How are the experience levels being affected by the turnover rates?
- What areas are most affected by turnover?

Measure - Earned Value

The Earned Value measure is a comparison between the cost of work performed and the budget, based on dollars budgeted per WBS element. The measure can be used to identify cost overruns and underpins.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Applicable to any project that uses a cost and schedule system such as a Cost Schedule Control System Criteria (C/SCSC) or an earned value measurement system • Useful during project planning, development and sustaining engineering phases **Process Integration** • C/SCSC data is required on most large DoD contracts, so it is often readily available. This data should be based on a validated cost accounting system. If this data is not required, then the cost measure can be used instead • This can be difficult to track without an automated system tied to the accounting system • This data tends to lag other measurement information due to formal reporting requirements • Limited in applicability if costs are planned and expended on a level of effort basis **Usually Applied During** • Requirements Analysis (Estimates and Actuals) • Design (Estimates and Actuals) • Implementation (Estimates and Actuals) • Integration and Test (Estimates and Actuals)	**Typical Data Items** • Budgeted Cost of Work Scheduled • Budgeted Cost of Work Performed • Actual Cost of Work Performed • Estimate at Completion • Budget at Completion **Typical Attributes** • Organization **Typical Aggregation Structure** • Software Activity **Typically Collected for Each** • Software Activity **Count Actuals Based On** • WBS element complete (to defined exit criteria) • WBS element percent complete (based on engineering judgment) • WBS element percent complete (based on underlying objective measures)

This Measure Answers Questions Such As:

- Are project costs in accordance with budgets?
- What is the projected completion cost?
- What WBS elements or tasks have the greatest variance?

Measure - Cost

The Cost measure counts budgeted and expended cost. The measure provides information about the amount of money expended on a project, compared to budgets.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Applicable to projects of all sizes and types. Used to evaluate costs for those projects that do not use cost schedule control system criteria (C/SCSC) • Useful during project planning, development, and sustaining engineering phases **Process Integration** • Data should come from an automated accounting system. This data tends to lag other measurement information due to formal reporting requirements • Should be relatively easy to collect at a high level. Not all projects, however, will break out software WBS elements to a sufficient level of detail • This measure does not address the amount of work completed for the costs incurred **Usually Applied During** • Requirements Analysis (Estimates and Actuals) • Design (Estimates and Actuals) • Implementation (Estimates and Actuals) • Integration and Test (Estimates and Actuals)	**Typical Data Items** • Cost (Dollars) **Typical Attributes** • Organization **Typical Aggregation Structure** • Software Activity **Typically Collected for Each** • Software Activity **Count Actuals Based On** • WBS element complete (to defined exit criteria) • WBS element percent complete (based on engineering judgment) • WBS element percent complete (based on underlying objective measures)

This Measure Answers Questions Such As:
- Are project costs in accordance with budgets?
- Will the target budget be achieved or will there be an overrun or surplus?

Measure - Resource Availability Dates

The Resource Availability Dates measure tracks the availability of key development and test environment resources. The measure is used to determine if key resources are available when needed. It can be integrated in the milestone dates measure.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • More important for projects with constrained resources • Useful during development and sustaining engineering phases **Process Integration** • Required data is generally easily obtained from project scheduling systems or documentation • Resources may include software, hardware, integration and test facilities, tools, other equipment, or office space. Normally only key resources are tracked. Personnel resources are not included in this measure - they are tracked with Effort • Be sure to consider all resources including those furnished by the government, the developer, and third party vendors **Usually Applied During** • Requirements Analysis(Estimates and Actuals) • Design (Estimates and Actuals) • Implementation (Estimates and Actuals) • Integration and Test (Estimates and Actuals)	**Typical Data Items** • Availability Date **Typical Attributes** None **Typical Aggregation Structure** • Software Activity **Typically Collected for Each** • Key Resource **Count Actuals Based On** • Demonstration of the intended service

This Measure Answers Questions Such As:
- Are key resources available when needed?
- Is the availability of support resources impacting progress?

Measure - Resource Utilization

The Resource Utilization measure counts the number of hours of resource time requested, allocated, scheduled, available, not available (due to maintenance downtime or other problems), and used. It is used on projects that have resource constraints, and is usually focused only on key resources. This measure provides an indication of whether key resources are sufficient and if they are used effectively.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • More important for projects with constrained resources. Especially important during integration and test activities • Useful during development and sustaining engineering phases **Process Integration** • Relatively easy to collect at a high level. Easier to collect if a resource monitor or resource scheduling system is in place • Resources may include software, hardware, integration and test facilities, tools, and other equipment. Normally only key resources are tracked • Include both Government-furnished and developer-furnished resources **Usually Applied During** • Requirements Analysis (Estimates and Actuals) • Design (Estimates and Actuals) • Implementation (Estimates and Actuals) • Integration and Test (Estimates and Actuals)	**Typical Data Items** • Requested Hours • Allocated Hours • Scheduled Hours • Available Hours • Hours Unavailable • Used Hours **Typical Attributes** • None **Typical Aggregation Structure** • Software Activity **Typically Collected for Each** • Key Resource **Count Actuals Based On** • End of reporting period

This Measure Answers Questions Such As:
- Are sufficient resources available?
- How efficiently are resources being used?

Measure – Lines of Code

The Lines of Code measure counts the total amount of source code and the amount that has been added, modified, or deleted. The total number of lines of code is a well understood measure that allows estimation of project cost, required effort, schedule, and productivity. Changes in the number of lines of code indicate development risk due to product size volatility and additional work that may be required.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains. Commonly used in weapons applications • Included in most DoD measurement policies and some commercial measurement practices • Used for projects of all sizes. Less important for projects where little code is generated such as those using automatic code generation and visual programming environments • Most effective for traditional high order languages such as Ada, FORTRAN, and COBOL. Not generally used for fourth-generation languages such as Natural and ECOS • Not usually tracked for COTS software unless changes are made to the source code • Useful during project planning, development, and sustaining engineering phases **Process Integration** • Define Lines of Code for each language. Lines of code from different languages are not equivalent • You may want to calculate an effective or equivalent Software Lines of Code count based on source. New and modified lines would count at 100% while reused code would count at a lower percentage (to address the effort required to integrate and test the reused code) • It is sometimes difficult to generate accurate estimates early in the project, especially for new types of projects • Estimates should be updated on a regular basis • Can be difficult estimating and tracking lines of code by source and type • Actuals can easily be counted using automated tools **Usually Applied During** • Requirements Analysis and Design (Estimates) • Implementation (Estimates and Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Number of Lines of Code • Number of Lines of Code Added • Number of Lines of Code Deleted • Number of Lines of Code Modified **Typical Attributes** • Version • Source (new, reused, NDI, GOTS, or COTS) • Language • Delivery Status (deliverable, non- deliverable) • End-Use Environment (operational, support) **Typical Aggregation Structure** • Component **Typically Collected for Each** • Unit or equivalent **Lines of Code Definition May Include** • Logical Lines • Physical Lines • Comments • Executables • Data Declarations • Compiler Directives **Count Actuals Based On** • Release to configuration management • Passing unit test • Passing inspection

This Measure Answers Questions Such As:

- How accurate was the size estimate on which the schedule and effort plans were based?
- How much has the software size changed? In what components have changes occurred?
- Has the size allocated to each incremental build changed? Is functionality slipping to later builds?

Chapter 7

Growth and Stability

Measure - Components

The Components measure counts the number of elementary software components in a software product, and the number that are added, modified, or deleted. The total number of components defines the size of the software product. Changes in the number of estimated and actual components indicate risk due to product size volatility and additional work that may be required. Reporting the number of components provides product size information earlier than other size measures, such as Lines of Code.

Selection Guidance	Specification Guidance
Project Application • Applicable to all application domains, generally with different component definitions • Applicable to all sizes and type projects • Not usually tracked for COTS software unless changes are made to the source code • Useful during development and sustaining engineering phases **Process Integration** • Requires a well-defined and consistent component allocation structure (i.e. unit to CI to build) • Required data is generally easy to obtain from software design tools, configuration management tools, or documentation • Deleted and added components are relatively easy to collect - modified components are often not tracked • Volatility in the planned number of components may represent instability in the requirements or in the design of the software **Usually Applied During** • Requirements Analysis (Estimates) • Design (Estimates and Actuals) • Implementation (Estimates and Actuals) • Integration and Test (Actuals)	**Typical Dab Items** • Number of Units • Number of Units Added • Number of Units Deleted • Number of Units Modified **Typical Attributes** • Version • Source (new, reused, NDI, GOTS, or COTS) • Language • Delivery Status (deliverable, non- deliverable) • End-Use Environment (operational, support) **Typical Aggregation Structure** • Component **Typically Collected for Each** • CI or equivalent **Count Actuals Based On** • Release to configuration management • Passing unit test • Passing inspection

This Measure Answers Questions Such As:

- How many components need to be implemented and tested?
- How much has the approved software baseline changed?
- Have the components allocated to each incremental build changed? Is functionality slipping to later builds

Measure - Words of Memory

This measure counts the number of words used in main memory, in relation to total memory capacity. This measure provides a basis to estimate if sufficient memory will be available to execute the software in the expected operational scenarios.

Selection Guidance	Specification Guidance
Project Application • Most commonly used for weapons systems • Used on any project with severe memory constraints such as avionics or on-board flight software • For many projects the amount of memory reserved is part of the defined exit criteria • Useful during development and sustaining engineering phases **Process Integration** • Requires an automated tool that measures usage based on a defined operational profile. This is often • Estimation may be based on modeling or by assuming a translation factor between lines of code and words of memory **Usually Applied During** • Requirements Analysis (Estimates) • Design (Estimates) • Implementation (Estimates and Actuals) • Integration and Test (Estimates and Actuals)	**Typical Data Items** • Number of Words of Memory • Number of Words of Memory Used **Typical Attributes** • Version **Typical Aggregation Structure** • Component **Typically Collected for Each** • Software CI or Hardware CI – Processor **Count Actuals Based On** • Release to configuration management • Passing unit test • Passing inspection • During Test Readiness Review • Prior to delivery

This Measure Answers Questions Such As:
- How much spare memory capacity is there?
- Does the memory need to be upgraded?

Measure - Database Size

The Database Size measure counts the number of words, records, or tables (elements) in each database. The measure indicates how much data must be handled by the system.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains. Often used for Automated Information System projects • Used for any project with a significant database. Especially important for those with performance constraints • Useful during development and sustaining engineering phases **Process Integration** • In order to estimate the size of a database, you must develop an operational profile. This is generally a manual process that can be difficult. Actuals are relatively easy to collect **Usually Applied During** • Requirements Analysis (Estimates) • Design (Estimates) • Implementation (Estimates and Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Number of Tables • Number of Records or Entries • Number of Words or Bytes **Typical Attributes** • Version **Typical Aggregation Structure** • Component **Typically Collected for Each** • Software CI – Database • Hardware CI – Processor **Count Actuals Based On** • Schema design released to configuration management • Schema implementation released to configuration management

This Measure Answers Questions Such As:
- How much data has to be handled by the system?
- How many different data types have to be addressed?

Measure- Requirements

The Requirements measure counts the number of requirements in the software and interface specifications. It also counts the number of these requirements that are added, modified, or deleted. The measure provides information on the total number of requirements, and the development risk due to volatility in requirements or functional growth.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Applicable to any project that tracks requirements • Useful for any size and type of project • Useful during project planning, development, and sustaining engineering phases • Effective for both non-developed (COTS/GOTS/Reuse) and newly developed software **Process Integration** • Requires a good requirements traceability process. If an automated design tool is used, the data is more readily available • Count changes against a baseline that is under formal configuration control. Both stated and derived requirements may be included • To evaluate stability, a good definition of the impacts of each change is required • It is sometime difficult to specifically define a "requirement." A consistently applied definition makes this measure more effective **Usually Applied During** • Requirements Analysis (Estimates and Actuals) • Design (Actuals) • Implementation (Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Number of Requirements • Number of Requirements Added • Number of Requirements Deleted • Number of Requirements Modified **Typical Attributes** • Version • Change Source (developer, acquirer, user) • Software Activity **Typical Aggregation Structure** • Function **Typically Collected for Each** • Requirement Specification **Count Actuals Based On** • Passing requirements inspection • Release to configuration management • Software Change Control Board Approval

This Measure Answers Questions Such As:
- Have the requirements allocated to each incremental build changed?
- Are requirements being deferred to later builds?
- How much has software functionality changed?
- What components have been affected the most?

Chapter 7

Growth and Stability

Measure- Function Points

The Function Points measure provides a weighted count of the number of external inputs and outputs, logical internal files and interfaces, and inquiries. This measure determines the functional size of software to support an early estimate of the required level of effort. It can also be used to normalize productivity measures and defect rates.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains. Commonly used in Automated Information System applications • Not usually tracked for COTS or reused software • Useful during development and sustaining engineering phases **Process Integration** • Requires a design process compatible with function points • Should be based on a defined method such as the IFPUG function point counting practices manual • Usually requires formal training • Requires a well-defined set of work products to describe the requirements and design • Very labor intensive to estimate and count - automated tools are scarce and have not been validated **Usually Applied During** • Requirements Analysis (Estimates) • Design (Estimates and Actuals) • Implementation (Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Number of Function Points • Number of Function Points Added • Number of Function Points Deleted • Number of Function Points Modified **Typical Attributes** • Version • Source (new, reused, NDI, GOTS, or COTS) **Typical Aggregation Structure** • Function • Component **Typically Collected for Each** • Function • CI or equivalent **Count Actuals Based On** • Completion of design documentation • Release to configuration management • Passing design documentation inspections • Delivery

This Measure Answers Questions Such As:

- How big is the software product?
- How much work is there to be done?
- How much functionality is in the software?

Measure - Change Request Workload

The Change Request Workload measure counts the number of change requests affecting a product. The measure provides an indication of the amount of work required and performed.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Applicable to all sizes of project • Useful during the development phase. Often used for projects in the sustaining engineering phase. Not generally used for integration projects incorporating COTS and reused code **Process Integration** • Data should be available for most projects • Often used on iterative developments including sustaining engineering projects doing basic maintenance **Usually Applied During** • Requirements Analysis (Actuals) • Design (Actuals) • Implementation (Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Number of Software Change Requests Written • Number of Software Change Requests Open • Number of Software Change Requests Assigned to a Version • Number of Software Change Requests Resolved **Typical Attributes** • Version • Priority • Valid/lnvalid • Approved/Unapproved • Change Classification (defect correction, enhancement) **Typical Aggregation Structure** • Function **Typically Collected for Each** • Requirement Specification • Design Specification **Count Actuals Based On** • Change submitted • Change approved • Change analyzed • Change implemented • Change integrated • Change tested

This Measure Answers Questions Such As:

- How many change requests have been written?
- Is the backlog of open change requests declining?
- Is the rate of new change requests increasing or decreasing?

Chapter 7

Product Quality

Measure- Problem Reports

The Problem Reports measure quantifies the number, status, and priority of problems reported. It provides very useful information on the ability of a developer to find and fix defects. The quantity of problems reported reflects the amount of development rework (quality). Arrival rates can indicate product maturity (a decrease should occur as testing is completed). Closure rates are an indication of progress and can be used to predict test completion. Tracking the length of time that problem reports have remained open can be used to determine whether progress is being made in fixing problems. It helps assess whether software rework is deferred.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Included in most DoD measurement policies and commercial measurement practices • Applicable to all sizes and types of projects • Useful during development and sustaining engineering phases **Process Integration** • Requires a disciplined problem reporting process. This measure is generally available during integration and test. It is beneficial, however, to begin problem tracking earlier during requirements, design, code, and unit test inspections and unit tests • The status codes used on a project should address at a minimum whether problem reports are open or resolved • Easy to collect actuals when an automated problem reporting system is used. Many projects do not estimate the number of problem reports expected • The number of discovered problem reports should be considered relative to the amount of discovery activity such as number of inspections and amount of testing • Many projects use the number of open problem reports, by priority categories, as a measure of readiness for test • To track age of problems reports, the project may collect average age, median age, longest age, or by age category (e.g. number open less than 1 month, 1-3 months, more than 3 months, etc.). Each project must define what activities are included in age (e.g. time from discovery to validation, integration, or field) **Usually Applied During** • Requirements Analysis (Estimates and Actuals) • Design, Integration and Test (Estimates and Actuals) • Implementation (Estimates and Actuals)	**Typical Data Items** • Number of Problem Reports • Average Age of Problem Reports **Typical Attributes** • Version • Priority • Problem Report Status Code • Software Activity Originated • Software Activity Discovered **Typical Aggregation Structure** • Component **Typically Collected for Each** • CI or equivalent **Count Actuals Based On** • Problem report documented • Problem report approved by configuration control board • Successfully tested • Successfully integrated • Delivery to field

This Measure Answers Questions Such As:

- How many (critical) problem reports have been written?
- Do problem report arrival and closure rates support the scheduled completion date of integration and test?
- How many problem reports are open? What are their priorities?

Measure - Defect Density

The Defect Density measure is a ratio of the number of defects written against a component relative to the size of that component. Either a product or function oriented size measure can be used. The measure helps identify components with the highest concentration of defects. These components often become candidates for additional reviews or testing, or may need to be re-written. Trends in the overall quality of a system can also be monitored with this measure.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Applicable to all sizes and types of projects • Useful during development and sustaining engineering phases **Process Integration** • Requires a disciplined problem reporting process and a method of measuring software size • Requires the allocation of defect and size data to the associated component affected • In order to use functional measures of size, requirements or function points must be allocated to the associated components • Actuals are relatively easy to collect. Most projects do not estimate defect density • Usually only valid, unique problem reports are included in the defect density calculation **Usually Applied During** • Requirements Analysis (Actuals) • Design (Actuals) • Implementation (Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Number of Defects • Number of Lines of Code **Typical Attributes** • Version • Priority • Source (new, reused, NDI, GOTS, or COTS) • Language **Typical Aggregation Structure** • Component **Typically Collected for Each** • CI or equivalent **Alternatives to Lines of Code Include** • Components • Requirements • Function Points **Count Actuals Based On** • Defects documented • Defects validated • Successfully integrated • Successfully tested • Delivered to field

This Measure Answers Questions Such As:
- What is the quality of the software?
- What components have a disproportionate amount of defects?
- What components require additional testing or review?
- What components are candidates for rework?

Measure - Failure Interval

The Failure Interval measure specifies the time between each report of a software failure. The measure is used as an indicator of the length of time that a project can be expected to run without a software failure (during systems operation). The measure provides insight into how the software affects overall system reliability. This measure can be used as an input to reliability prediction models.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Applicable to any project with reliability requirements • Useful during development in system or operational test. Used throughout sustaining engineering based on reported operational failures **Process Integration** • Requires a disciplined failure tracking process. Easier to collect if an automated system is used. Data can be gathered from test logs or incident reports • Consider what priority of failures to include • Be sure to exclude non-software failures. This includes failures caused by hardware problems as well as user generated failures caused by operator error or user documentation errors • Some projects specify threshold limits on an acceptable number of failures per operating time for software reliability • Consider whether or not to count duplicate failures. • Consider how to count operational time on interfacing hardware **Usually Applied During** • Integration and Test (Actuals)	**Typical Data Items** • Failure Identifier • Failure Date/Time Stamp • Operating Time to Failure **Typical Attributes** • Version • Failure Priority **Typical Aggregation Structure** • Component **Typically Collected for Each** • Hardware CI • Software CI **Count Actuals Based On** • Failure documented • Failure validated

This Measure Answers Questions Such As:

- What is the project's expected operational reliability?
- How often will software failures occur during operation of the system?
- How reliable is the software?

Measure – Cyclomatic Complexity (Logic Paths)

The Cyclomatic Complexity measure counts the number of unique logical paths contained in a software component. This measure helps assess both code quality and the amount of testing required. A high complexity rating is often indicative of a high defect rate. Components with high complexity usually require additional reviews or testing, or may need to be re-written.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Applicable to projects with testability, reliability, or maintainability concerns • Not generally used for COTS or reused code. Not generally used on software from automatic code generators or visual programming environments • Useful during development and sustaining engineering phases. **Process Integration** • Cyclomatic complexity does not differentiate between type of control flow. A CASE statement counts as high complexity even though it is easier to use and understand than a series of conditional statements • Cyclomatic complexity does not address data structures • Operational requirements may require efficient, highly complex code • Relatively easy to collect actuals when automated tools are available (e.g., for Ada, C, C++). Estimates are generally not derived, but a desired threshold or expected distribution may be specified **Usually Applied During** • Design (Actuals) • Implementation (Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Cyclomatic Complexity Rating **Typical Attributes** • Version **Typical Aggregation Structure** • Component **Typically Collected for Each** • Unit or equivalent **Count Actuals Based On** • Passing inspection • Passing unit test • Release to configuration management

This Measure Answers Questions Such As:
- How many complex components exist in this project?
- What components are the most complex?
- What components should be subject to additional testing?
- What is the minimum number of reviews and test cases required to test the logical paths through the component?

Measure - Rework Size

The Rework Size measure counts the number of lines of code changed to fix known defects. This measure helps in assessing the quality of the initial development effort, by indicating the amount of total code that had to undergo rework.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Applicable to most development processes. In a rapid prototype process, it is only applicable to the "final" version of the software product • Not generally used for non-developed code such as COTS • Useful during development and sustaining engineering phases **Process Integration** • Very difficult to collect. Most configuration management systems do not collect information on changes to the size of code or reason for the change (rework) • Rework size should only include code changed to correct defects. Changes due to enhancements are not rework • Rework cost and schedule estimates should be included in the development plan **Usually Applied During** • Implementation (Actuals) • Integration and Test (Actuals) •	**Typical Data Items** • Number of Lines of Code added due to rework • Number of Lines of Code deleted due to rework • Number of Lines of Code modified due to rework **Typical Attributes** • Version • Language • Delivery Status (deliverable, non- deliverable) • End-Use Environment (operational, support) **Typical Aggregation Structure** • Component **Typically Collected for Each** • Unit or equivalent **Alternatives to Lines of Code Include** • Components **Count Actuals Based On** • Release to configuration management • Passing inspection • Passing unit test

This Measure Answers Questions Such As:
- How much code had to be changed as a result of correcting defects?
- What was the quality of the initial development effort?
- Is the amount of rework impacting the cost and schedule?

Measure- Rework Effort

The Rework Effort measure counts the amount of work effort expended to find and fix software defects. Rework effort may be expended to fix any software product, including those related to requirements analysis, design, code, etc. This measure helps assess the quality of the initial development effort, and identify products and software activities requiring the most rework.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Applicable to most development processes. In a rapid prototype process, it is only applicable to the "final" version of the software product • Not generally used for effort associated with non-developed code such as COTS • Useful during development and sustaining engineering phases **Process Integration** • Difficult to collect. Some cost accounting systems do not collect information on rework effort • For basic tracking, a single WBS/cost account should be created to track all rework effort (per organization). For more advanced tracking, multiple WBS/cost accounts should be created to track rework at the component and/or activity level • Rework effort should only include effort associated with correcting defects. Effort expended due to incorporation of enhancements is not rework • Rework cost and schedule estimates should be included in the development plan **Usually Applied During** • Requirements Analysis (Actuals) • Design (Actuals) • Implementation (Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Labor Hours Due to Rework **Typical Attribute** • Organization • Labor Category • Version • Software Activity **Typical Aggregation Structure** • Software Activity **Typically Collected for Each** • Software Activity **Count Actuals Based On** • End of financial reporting period

This Measure Answers Questions Such As:

- How much effort was expended on fixing defects in the software product?
- What software activity required the most rework?
- Is the amount of rework impacting cost and schedule?

Measure - Capability Maturity Model Level

The Capability Maturity Model (CMM) Level measure reports the rating (1-5) of a software development organization's software development process, as defined by the Software Engineering Institute. The measure is the result of a formal assessment of the organization's project management and software engineering capabilities. It is often used during the source selection process to evaluate competing developers.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Normally measured at the organizational level • Useful during project planning, development and sustaining engineering phases **Process Integration** • Requires formal training and a very structured assessment approach. Requires a significant amount of time and effort • An external assessor may formally conduct an assessment, or a self-evaluation can be performed • Rating may be used during source selection to help select a developer. Assessment may be used as part of a process improvement project **Usually Applied During** • Not applicable	**Typical Data Items** • CMM Rating **Typical Attributes** • None **Typical Aggregation Structure** • Software Activity (Organization) **Typically Collected for Each** • Organization **Count Actuals Based On** • Prior to contract award • External or Self Evaluation

This Measure Answers Questions Such As:
- Does a developer meet minimum development capability requirements?
- What is the developer's current software development capability?
- What project management and software engineering practices can be improved?
- Is the developer's software process adequate to address anticipated project risks, issues, and constraints?

Measure - Product Size/Effort Ratio

The Product Size/Effort Ratio measure specifies the amount of software product produced relative to the amount of effort expended. This common measure of productivity is used as a basic input to project planning and also helps evaluate whether performance levels are sufficient to meet cost schedule estimates.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains. Commonly used in weapons systems • Used for projects of all size. Less important for projects where little code is generated such as those using automatic code generation and visual programming environments • Not generally used for COTS or reused software • Estimates are often used during project planning. Both estimates and actuals are used during development and sustaining engineering to focus on the incorporation of new functionality. Not generally used for maintenance projects focused on problem resolution **Process Integration** • In order to compare productivity from different projects, the same definitions of size and effort must be used. For size, the same measure (e.g., Lines of Code) must be used as well as the same definition (e.g., Logical lines). For the effort measure, the same labor categories and software activities must be included • The environment, language, tools, and personnel experience will effect productivity achieved • Productivity can also be calculated using software cost models. Many of these models include schedule as part of the productivity equation • To validly calculate productivity, the effort measure must correlate directly with the size measure. If, for example, effort for a component is included but the component's size is not, productivity will be lower • Definitions should specify those elements of effort that are included (e.g., project management, documentation, etc.) **Usually Applied During** • Requirements Analysis (Estimates and Actuals) • Design (Estimates and Actuals) • Implementation (Estimates and Actuals) • Integration and Test (Estimates and Actuals)	**Typical Data Items** • Number of Lines of Code • Number of Labor Hours **Typical Attributes** • Version • Language **Typical Aggregation Structure** • Software Activity **Typically Collected for Each** • Organization **Alternatives to Lines of Code Include** • Components • Tables • Records or Entities **Alternatives to Labor Hours Include** • Labor Days/Weeks/Months • Full Time Equivalents • Number of Personnel **Count Actuals Based On** • Completion of Version • Components implemented • Components integrated and tested

This Measure Answers Questions Such As:
- Is the developer's production rate sufficient to meet the completion date?
- How efficient is the developer at producing the software product?
- Is the planned/required software productivity rate realistic?

Chapter 7

Development Performance

Measure - Functional Size/Effort Ratio

The Functional Size/Effort Ratio measure specifies the amount of functionality produced relative to the amount of effort expended. This measure is used as a basic input to project planning and also helps evaluate whether performance levels are sufficient to meet cost schedule estimates.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains. Commonly used in AIS systems • Useful when product size measures are not available • Useful during project planning, development, and sustaining engineering phases **Process Integration** • In order to compare productivities from different projects, the same definitions of size and effort must be used. For size, the same measure (e.g. Function Points) must be used as well as the same counting practices. For the effort measure, the same labor categories and software activities must be included • The environment, language, tools, and personnel experience will effect productivity achieved • Productivity can also be calculated using software cost models. Many of these models include schedules as part of the productivity equation • To validly calculate productivity, the effort measure must correlate directly with the size measure. If, for example, effort for a function is included but the functional size is not, productivity will be lower • Useful early in the project, before actual product size data is available **Usually Applied During** • Requirements Analysis (Estimates and Actuals) • Design (Estimates and Actuals) • Implementation (Estimates and Actuals) • Integration and Test (Estimates and Actuals)	**Typical Data Items** • Number of Requirements • Number of Labor Hours **Typical Attributes** • Version **Typical Aggregation Structure** • Software Activity **Typically Collected for Each** • Organization **Alternatives to Requirements Include** • Function Points **Alternatives to Labor Hours Include** • Labor Days/Weeks/Months • Full Time Equivalents • Number of Personnel **Count Actuals Based On** • Completion of Version • Functions implemented • Functions integrated and tested

This Measure Answers Questions Such As:

- Is the developer producing the software at a sufficient rate to meet the completion date?
- How efficient is the developer at producing the software?
- Is the planned/required software productivity rate realistic?

Measure - CPU Utilization

The CPU Utilization measure counts the estimated or actual proportion of time the CPU is busy during a measured time period. This measure indicates whether sufficient CPU resources will be available to support operational processing. This measure is also used to evaluate whether CPU reserve capacity will be sufficient for high-usage operations or for added functionality.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains. Primarily used for weapon systems • Useful for any project with a dedicated processor and critical performance requirements. Not generally used on projects located on shared processors • Useful during development and sustaining engineering phases **Process Integration** • Requires a tool that measures usage based on a defined operational profile during a measured period of time • The operational profile (load levels) has a significant impact on this measure. Test should include both normal and stress levels of operation. The operational profile for each test should be provided with the data • Estimates are very difficult to derive and require significant simulation or modeling support. Estimates must be developed early to impact design decisions • Actual processor utilization is often provided as an overhead function of an operating system and is more easily obtained **Usually Applied During** • Design (Estimates) • Implementation (Estimates and Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Time Processor is Busy • Measured Time Period • Specified Processor Utilization Limit **Typical Attributes** • Version • Operational profile **Typical Aggregation Structure** • Component **Typically Collected for Each** • Hardware CI – Processor **Count Actuals Based On** • Integrated system test • Stress/endurance test

This Measure Answers Questions Such As:
- Have sufficient CPU resources been provided?
- Do CPU estimates appear reasonable? Have large increases occurred?
- Can the CPU resources support additional functionality?

Measure - CPU Throughput

The CPU Throughput measure provides an estimate or actual count of the number of processing tasks that can be completed in a specified period of time. This measure provides an indication of whether or not the software can support the system's operational processing requirements.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains. Primarily used for weapon systems • Useful for any project with a dedicated processor and critical timing requirements. Not generally used on projects located on shared processors • Useful during development and sustaining engineering phases **Process Integration** • Actuals can be based on real-time observation or may require a tool that measures task completion based on a defined operational profile. This data is generally easy to collect • The operational profile has a significant impact on this measure. Tests should include both normal and stress levels of operation. The operational profile for each test should be provided with the data • Estimates are very difficult to derive and require significant simulation or modeling support. • Estimates must be developed early to impact design decisions • The measurement methodology for CPU throughput is critical for meaningful results. In many cases the measure is based on average CPU throughput. The averaging period used is therefore important **Usually Applied During** • Design (Estimates) • Implementation (Estimates and Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Number of Requests for Service • Number of Requests for Service Completed • Measured Time Period • Specified Processor Throughput Limit **Typical Attributes** • Version • Operational Profile **Typical Aggregation Structure** • Component **Typically Collected for Each** • Hardware CI – Processor **Count Actuals Based On** • Integrated system test • Stress/endurance test

This Measure Answers Questions Such As:

- Have sufficient CPU resources been acquired?
- Do CPU estimates appear reasonable? Have large increases occurred?

Measure -I/0 Utilization

The l/O Utilization measure calculates the proportion of time the l/O resources are busy during a measured time period. This measure indicates whether l/O resources are sufficient to support operational processing requirements.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains. Primarily used for weapon systems • Critical for high traffic systems • Network l/O may also be measured under this measure • Useful during development and sustaining engineering phases **Process Integration** • Actual measurement requires a tool that measures usage based on a defined operational profile during a measured period of time. Actuals are relatively easy to collect • The operational profile has a significant impact on this measure. The test cases should include both normal and stress levels of operation. The operational profile for each test should be provided with the data • Estimates are very difficult to derive and require significant simulation or modeling support. • Estimates must be developed early to impact design decisions **Usually Applied During** • Design (Estimates) • Implementation (Estimates and Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Time l/O Resource is Busy • Time l/O Resource is Available • Measured Time Period • Specified l/O Channel Utilization Limit **Typical Attributes** • Version • Operational Profile **Typical Aggregation Structure** • Component **Typically Collected for Each** • Hardware CI - I/O Device **Count Actuals Based On** • Integrated system test • Stress/endurance test

This Measure Answers Questions Such As:

- Do the l/O resources allow adequate data traffic flow?
- Can additional data traffic be provided after system delivery?
- Should l/O resources be expanded?

Measure -I/0 Throughput

The l/O Throughput measure reports the rate at which the l/O resources send and receive data, according to the number of data packets (bytes, words, etc.) successfully sent or received during a measured time period. This measure indicates whether the l/O resources are sufficient to support the system's operational processing requirements.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains. Primarily used for weapon systems • Critical for high traffic systems • Network l/O may also be measured under this measure • Useful during development and sustaining engineering phases **Process Integration** • Actual measurement requires a tool that measures usage based on a defined operational profile during a measured period of time. This is relatively easy to collect • The operational profile has a significant impact on this measure. Tests should include both normal and stress levels of operation. The operational profile for each test should be provided with the data • Estimates are very difficult to derive and require significant simulation or modeling support. • Estimates must be developed early to impact design decisions • The measurement methodology for l/O throughput is critical for meaningful results. In many cases the measure is based on average l/O throughput, therefore, the averaging period used is very important **Usually Applied During** • Design (Estimates) • Implementation (Estimates and Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Number of Data Packets • Number of Data Packets Successfully Sent • Number of Data Packets Successfully Received • Measured Time Period • Specified l/O Throughput Limit **Typical Attributes** • Version • Operational Profile **Typical Aggregation Structure** • Component **Typically Collected for Each** • Hardware CI - I/O Device **Count Actuals Based On** • Integrated system test • Stress/endurance test

This Measure Answers Questions Such As:

- Can the software design handle the required amount of system data in the allocated time?
- Can the software handle additional system data after delivery?

Measure- Memory Utilization

The Memory Utilization measure indicates the proportion of memory that is used during a measured time period. This measure addresses random access memory (RAM), read only memory (ROM), or any other form of electronic, volatile memory. This measure specifically excludes all types of magnetic and optical media (e.g. disk, tape, CD-ROM, etc.). This measure provides an indication of whether the memory resources can support the system's operational processing requirements.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains. Primarily used for weapon systems • Critical for memory constrained systems • Useful during development and sustaining engineering phases **Process Integration** • Measure and monitor different types of memory (e.g., RAM, ROM) separately. Specify the size of a word (e.g., 16 bit, 32 bit, etc.) for each memory type. • Actual measurement requires a tool that measures usage based on a defined operational profile during a measured time period or task. This is relatively easy to collect • The operational profile has a significant impact on this measure. The tests should include both normal and stress levels of operation. The operational profile for each test should be provided with the data • Estimates are very difficult to derive and require significant simulation or modeling support. • Estimates must be developed early to impact design decisions **Usually Applied During** • Design (Estimates) • Implementation (Estimates and Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Memory • Memory Available • Memory Used • Measured Time Period • Specified Memory Utilization Limit **Typical Attributes** • Version • Operational Profile **Typical Aggregation Structure** • Component **Typically Collected for Each** • Hardware CI – Processor **Count Actuals Based On** • Integrated system test • Stress/endurance test

This Measure Answers Questions Such As:
- Will the software fit in the processors?
- Can the software size increase after system delivery as needed to incorporate new functionality?
- What is the risk that system errors will be caused by lack of storage space?

Measure – Storage Utilization

The Storage-Utilization measure reports the proportion of storage capacity used. The measure provides an indication of whether storage resources are sufficient to store projects and/or the anticipated volume of operational data generated by the system. The term "storage" refers to magnetic and optical media (e.g. disk, tapes, hard drives, CD-ROM, etc.), but specifically excludes all types of random access memory (RAM), read only memory (ROM), or any other forms of electronic memory

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains. Primarily used for weapon systems • Critical for storage constrained systems • Useful during development and sustaining engineering phases **Process Integration** • Measure and monitor different types of storage (e.g., disk, tape) separately. Specify the size of a word (e.g., 16 bits, 32 bits, etc.) for each storage type • Actuals are easy to measure. Estimates are often based on product size **Usually Applied During** • Design (Estimates) • Implementation (Estimates and Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Storage • Storage Available • Storage Used • Specified Storage Utilization Limit **Typical Attributes** • Version **Typical Aggregation Structure** • Component **Typically Collected for Each** • Hardware CI - Storage Unit **Count Actuals Based On** • Integrated system test • Stress/endurance test

This Measure Answers Questions Such As:

- Have sufficient storage resources been provided?
- Do storage estimates appear adequate?
- What is the expansion capacity?

Measure - Response Time

The Response Time measure reports the amount of time required to process a request. The measure counts the time between initiation of a request for service and the conclusion of that service. It provides an indication that the target computer system responds in a timely manner. User interface response time is often considered an important quality factor.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains. Used extensively on AIS systems • Critical for projects with specified response time requirements. Especially critical for real-time projects • Useful during development and sustaining engineering phases **Process Integration** • Actuals can be based on real-time observation or may require a tool that measures request completion based on a defined operational profile. This data is generally easy to collect • The operational profile has a significant impact on this measure. Tests should include both normal and stress levels of operation. The operational profile for each test should be provided with the data • This measure must be collected at a low level in order to provide a good characterization of the level of service provided **Usually Applied During** • Design (Estimates) • Implementation (Estimates and Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Service Initiation Time • Service Completion Time • Maximum Allowable Service Time **Typical Attributes** • Version • Operational Profile **Typical Aggregation Structure** • Function **Typically Collected for Each** • Function – Service **Count Actuals Based On** • Integrated system test • Stress/endurance test

This Measure Answers Questions Such As:
- Is the target computer system sufficient to meet response requirements?
- How long do certain services take?
- Does the software operate efficiently?

Measure--Achieved Accuracy in Software Performance

The measure of Achieved Accuracy in Software Performance is usually a combination of several other measures that are defined by the software functional and technical requirements. These measures can include any functional characteristics that can be quantitatively defined and demonstrated during the software or system operation. Technical Performance measures are usually defined in term of the accuracy of the functions of the software or system to meet defined requirements, such as response time, data handling capability, or signal processing. These measures provide an indication of the overall ability of a software- intensive system to meet the users' functional requirements.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains • Included in all government and commercial projects that define specific requirements that must be achieved in software products • Used for projects of all sizes • Often used for projects integrating COTS software • Useful during development and sustaining engineering phases **Process Integration** • Sometimes difficult to generate accurate estimates early in the project, especially for new technologies and new projects • Data may not be available until late in a project, when system functional testing is performed • Resource and technology limitations may prohibit demonstration and measurement of all technical performance parameters • Data is usually available from functional test records • Modeling and simulation results may be used to estimate software functional performance levels **Usually Applied During** • Requirements Analysis (Estimates) • Design (Estimates) • Implementation (Estimates and Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Software functional performance level **Typical Attributes** • Version • Source (new, reused, NDI, GOTS, or COTS) **Typical Aggregation Structure** • Component **Typically Collected for Each** • CI or equivalent **Count Actuals Based On** • Passing functional test

This Measure Answers Questions Such As:
- How accurate was the signal processing function in this software release?
- Is the system able to read all the required data files in the available time?
- Was the software able to perform all required functions to meet the required system response time?

Measure- NDI Utilization

The NDI Utilization measure tracks the amount of code that is planned for reuse against what is actually reused. If less code is reused than planned, additional schedule and effort will most likely be required to complete the development.

Selection Guidance	Specification Guidance
Project Application • Applicable to all domains. Commonly used in weapons applications • Included in most DoD measurement policies and some commercial measurement practices • Used for projects of all sizes. Less important for projects where little code is generated such as those using automatic code generation and visual programming environments • Most effective for traditional high order languages such as Ada, FORTRAN, and COBOL. Not generally used for fourth- generation languages such as Natural and ECOS • Not usually tracked for COTS software unless changes are made to the source code • Useful during project planning, development, and sustaining engineering phases **Process Integration** • Define Lines of Code for each language. Lines of code from different languages are not equivalent • Sometimes difficult to generate accurate estimates early in the project, especially for new types of projects • Estimates should be updated on a regular basis. • Can be difficult estimating and tracking lines of code by source (new, modified, deleted, reused, NDI, GOTS, or COTS) • Actuals can easily be counted using automated tools **Usually Applied During** • Requirements Analysis (Estimates) • Design (Estimates) • Implementation (Estimates and Actuals) • Integration and Test (Actuals)	**Typical Data Items** • Number of Lines of Code (LOC) **Typical Attributes** • Version • Source (new, reused, NDI, GOTS, or COTS) • Type (added, deleted, modified) • Language **Typical Aggregation Structure** • Component **Typically Collected for Each** • Unit or equivalent **Alternatives to Lines of Code Include** • Components • Function Points • Requirements **Lines of Code Definition May Include** • Logical Lines • Physical Lines • Comments • Executables • Data Declarations • Compiler Directives **Count Actuals Based On** • Release to configuration management • Passing unit test • Passing inspection

This Measure Answers Questions Such As:
- How accurate was the reuse size estimate on which the schedule and effort plans were based?
- How much has the reuse software size changed? In what components have changes occurred?
- Has the reuse size allocated to each incremental build changed?

Implementing the Measurement Process

Once the project has begun, the analysis of software measures becomes a major concern. Analysis is conducted to determine whether software development efforts are meeting defined plans, assumptions and targets. Planned and actual performance data are the inputs to this process. Performance analysis should be viewed as an investigative process used to identify risks, manage risks and track down and isolate problems. This may require the use of slightly different data, the use of different measures to generate different indicators and the identification of alternative courses of action each time performance is analyzed.

Many times schedule, resources, growth, or quality trends are not recognizable as an indication of a potential problem until the associated risk has actually become a problem of major proportions. Because software risks are not independent, an integrated analysis using multiple indicators should be performed. In combination, Figures 7-1 and 7-2 show an example of a potential problem made visible by detecting inconsistent trends using multiple indicators. Figure 7-1 shows an indicator for the measure "Component Status" during the design process and Figure 7-2 shows an indicator for the measure "Problem Report Status" for the same project. Whereas the measure of actual component status appears to be only slightly behind the plan, the discrepancy between the number of open and closed problem reports is increasing. These open problem reports represent rework that must be completed before the design activity can be completed. Thus, the trends in these two performance indicators are inconsistent, an indication of a potential problem.

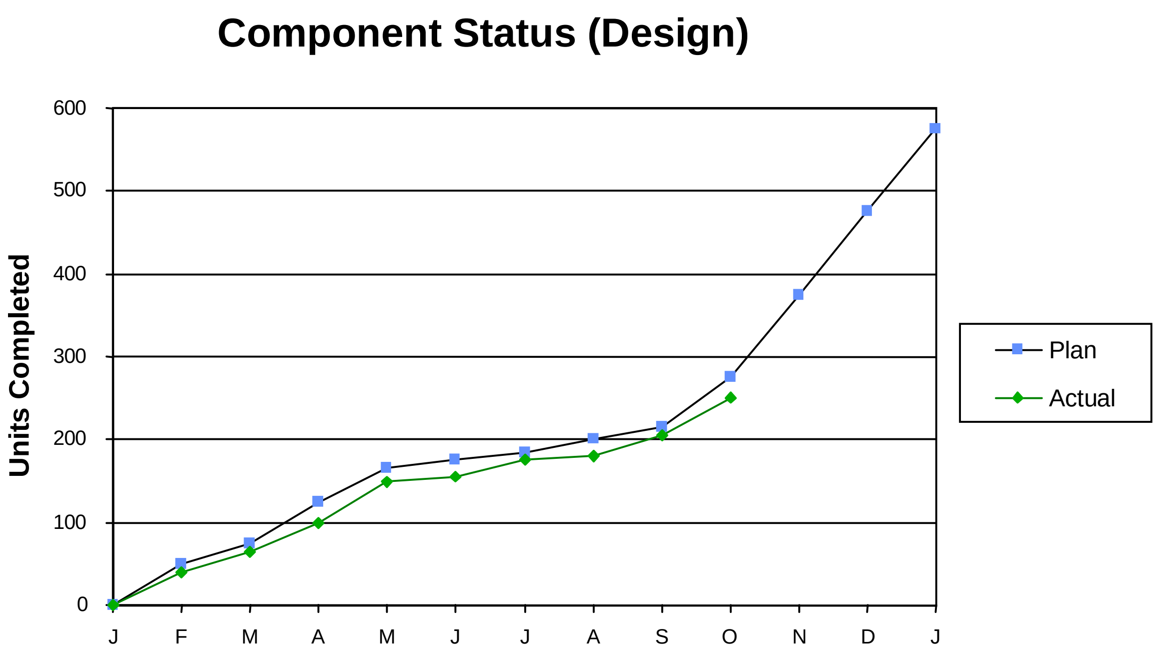


Figure 7-1. Component Status Indicator Example

Once a potential problem has been identified, it should be localized by examining indicators with more detailed data. In the example just cited, a Problem Report Status chart should be generated for each of the Configuration Items within the software design. Identifying the specific source of the potential problem helps to determine the rot cause and selection of the appropriate corrective action(s).

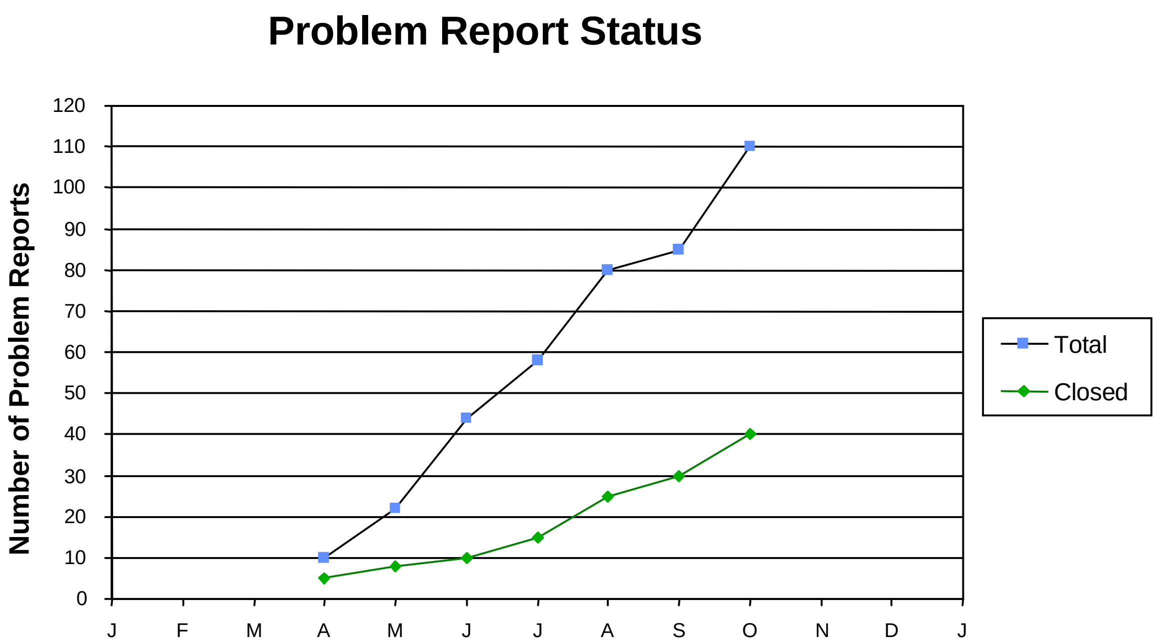


Figure 7-2. Problem Report Status Indicator Example

Watch-Out-Fors

The following examples of lessons learned were extracted from the publication "Practical Software Measurement, A Foundation for Objective Program Management." This list is provided as a starting point and is by no means comprehensive. The list is organized by common software risk categories.

Schedule and Progress

- A >10% cumulative, or >20% per period, actual deviation from planned progress (Once an actual progress trend line is established, it is difficult to change the rate of completion)
- A 5% or greater build schedule variance for single builds or a 10% build schedule variance across two or more builds

Resources and Cost

- Voluntary staff turnover >10% per year
- Large overruns during integration and test, which may indicate quality problems with the code and significant defects that may delay completion
- The addition of large numbers of people within a short period of time (this normally cannot be done effectively)

Growth and Stability

- Total software size increases > 20% over original estimates
- Constantly changing requirements or a large number of additions after requirements reviews, which are leading indicators of schedule and budget problems later in the project

Product Quality

- Defect removal efficiency <85%
- Large gaps between the closure rate and the discovery rate, indicating that problem correction is being deferred, which could result in serious schedule, staffing and cost problems later in the project
- A horizontal problem discovery trend line during design, coding or testing. This may indicate that reviews and tests are not being performed and should be investigated

Development Performance

- A developer with a poor software development process or low productivity, coupled with aggressive project schedule and cost objectives
- Unplanned rework, which is a frequent cause of low productivity
- Attempts to increase productivity significantly on an existing project

Technical Adequacy

- Changes in assumptions concerning the use of COTS software or the amount of code that can be reused

Additional information concerning software measurement may be found in the publication "Practical Software Measurement, A Foundation for Objective Project Management," Version 3.1, 17 April 1998 sponsored by the Joint Logistics Commanders Joint Group on Systems Engineering, and at:

http://www.psmsc.com

Chapter 8

Assess, Mitigate, Report...

What is the Relationship Between *"Assess, Mitigate, Report..."* and Technical Risk?

The risk assessment, mitigation, and reporting process often is not well structured and lacks discipline in its implementation. For example, assessments that reflect high risks are not encouraged, reporting is not done at the necessary level, and program schedules dictate the "what" and "when" of risk mitigation. When the disciplines in this process fail, resources and management attention cannot be applied to resolve risk issues and consequently, corrective actions remain open. The assessment, mitigation, and reporting of risk are the heart of the risk process and when this is not effective, the risk management program fails. Remember that a risk is not a problem, a problem is a risk that has already occurred.

Risk Assessment

The starting point for determining total program risk is to identify known or potential risk areas. This responsibility should be accepted by each person involved with the design, test, construction/manufacture, operation, support and eventual disposal of a weapon system, its subsystems and components. The earlier in the program that these risks are identified, the easier they will be to manage and the less negative impact they will have on program cost, schedule, and performance objectives. To facilitate the proactive identification of risks, there are methods and tools available for consideration. See Chapter 1, Figure 1-1, *Critical Process Risk Management*.

The next step is to assess the technical risk of each risk area identified within the program. These risk assessments are conducted to determine the disruption to your program as a function of two parameters, the level of each critical process variance from a known standard, and the consequence, i.e., the magnitude of the impact if the particular risk is realized. The levels for each parameter are used to enter the Assessment Guide grid, as shown in Chapter 1, Figure 1-2, *Critical Process Risk Assessment*, and the result is either a low, moderate, or high risk assessment. Note that the "Consequence" level takes into account the impact on technical performance, schedule, cost and other teams.

Total program risk assessment is determined by "rolling-up" all of the critical process risk area assessments affecting the program. The approach to risk assessment may vary depending on program philosophy. Chapter 1, "Choose an Approach," provides additional details on approaches to technical risk management.

Conducting Assessments

Assessments should be conducted in a manner that both optimizes program resources and schedule, and at the same time, is proactive in identifying risks before they become major program problems. Assessments should expose the potential weaknesses of the program – and therefore, should be conducted by subject matter experts from affected areas. There are three types of risk assessment:

- **Periodic Assessments**: Risk assessments are conducted at predetermined intervals, normally in preparation for milestone reviews. This approach may be sufficient for programs with limited resources, however, with this approach, low risks could develop into higher program risks if not identified early enough.
- **Continuous Assessments**: Risk assessments are on-going activities conducted by teams, rather than activities conducted only at scheduled times, such as program milestones, major events, etc. This is a proactive approach, allowing program risks to be identified early and mitigation strategies to be developed before technical risks impact performance, cost, and schedule. Continuous assessments are especially beneficial during the early phases of a program's life cycle.
- **Independent Risk Assessments**: Risk assessments are conducted by an outside team of experts, with experts normally coming from other programs or from industry. This is a recommended practice, as the assessors provide an unbiased review of the program and draw on their particular expertise to assess program risk. This is such an effective tool, that it is further discussed in Chapter 9, "Use Independent Assessors."

Experience to date, however, indicates that continuous assessments, coupled with independent assessments when necessary, represent the most effective strategy for assessing program risk.

PM Note

Evaluating Critical Process Variance

For each potential risk identified, the question must be asked: "W*hat is the Critical Process Variance from known standards or best practices?*" Looking at Chapter 1, Figure 1-2, there are five choices (levels) of Critical Process Variance: *Minimal, Small, Acceptable, Large, and Significant*. Associated with these five "levels" are the letters 'a' through 'e'. They correspond to the y-axis on the Assessment Guide. If the variance of a process from a known standard or best practice is considered minimal (level 'a'), the risk will be determined by proceeding along the 'a' row to the Consequence level selected. The risk will be low per this figure, unless of course the Consequence is considered "Significant" (level '5').

Evaluating Consequence

Risk Consequence is evaluated by answering the following question: *"Given the identified risk is realized, what is the magnitude of the impact of that risk?"* Levels of Consequence are labeled 1 through 5 and correspond to the x-axis on the Assessment Guide.

"Consequence" is a multifaceted issue. Applicable consequences have been narrowed down to four key areas (again referring to Chapter 1, Figure 1-2): Technical Performance, Schedule, Cost, and Impact on Other Teams. At least one (maybe more) of the four consequence metrics needs to apply for there to be the potential for risk. However, if there is no adverse Consequence, there is no risk irrespective of the assessed level of Critical Process Variance. These four metrics are further discussed as follows:

- ***Technical Performance***: The wording of each level is oriented toward design processes, but it should be applied as well to test processes, production processes, life cycle support, and equipment disposal. For example, the word "margin" could apply to weight margin during design, safety margin during testing, or machine performance margins during construction/manufacture and subsequent life cycle operation.

- ***Schedule***: The words used in the *Schedule* column, as in all columns of the Consequence Table, are meant to be generic. Avoid excluding a consequence level from consideration just because it doesn't match a team's specific definitions.

- ***Cost***: Cost is considered an *independent variable* in Defense programs. Since the magnitude of the dollars varies from component to component and process to process, percentage of dollars is used. This is also in step with Acquisition Program Baseline objectives and threshold values, which are based on percentage of dollars. The levels listed here represent costs at the program level. However, Integrated Product Teams (IPTs) may choose to align these definitions with standard cost reporting requirements consistent with cost consequences faced at the lower levels. At the program level, the definitions are as follows: Level 1 is Minimal or No Impact, Level 2 is <5%, Level 3 is 5 to 7%, Level 4 is >7 to 10%, and Level 5 is >10%.

- ***Impact on Other Teams***: Both the consequence of a risk and the mitigation actions associated with reducing risk may impact another IPT. When this impact results in increased complexity, levels of risk also increase. This may involve additional coordination or management attention (resources) and may therefore increase the level of risk.

Even after the Process Variance and Consequence levels have been determined, classification of the level of risk can be somewhat subjective, e.g., it could depend on the type of data being assessed. However, all assessments should be based on *experienced* judgment from your *best technical* people. Figures 1-1 and 1-2 of Chapter 1, discussed previously, provide a risk management and assessment tool based upon a process oriented approach, and ***Critical Process Variances*** from known best practice standards are plotted against ***Consequences*** to derive a level of risk assessment.

Another approach used by some programs is provided for information in Figure 8-1, in which values of ***Probability (or Likelihood) of Occurrence*** and ***Consequence*** are assigned to each risk element, with probability or likelihood plotted against consequence to derive a risk assessment level.

Likelihood

What is the Likelihood the Risk Will Happen?		
Level		Your Approach and Processes...
1	Not Likely:	...Will effectively avoid or mitigate this risk base on standard practices
2	Low Likelihood:	...Have usually mitigated this type of risk with minimal oversight in similar cases
3	Likely:	...May mitigate this risk, but workarounds will be required
4	Highly Likely:	...Cannot mitigate this risk, but a different approach might
5	Near Certainty:	...Cannot mitigate this type of risk; no known processes or workarounds are available

Consequence

Given the risk is realized, what would be the magnitude of the impact?			
Level	Technical	Schedule	Cost
1	Minimal or no impact	Minimal or no impact	Minimal or no impact
2	Minor perf. shortfall, same approach retained	Additional activities required; able to meet key dates	Budget increase or unit production cost increase <1%
3	Mod perf. shortfall, but workarounds available	Minor schedule slip; will miss need date	Budget increase or unit production cost increase <5%
4	Unacceptable, but workarounds available	Program critical path affected	Budget increase or unit production cost increase <10%
5	Unacceptable; no alternatives exist	Cannot achieve key program milestone	Budget increase or unit production cost increase >10%

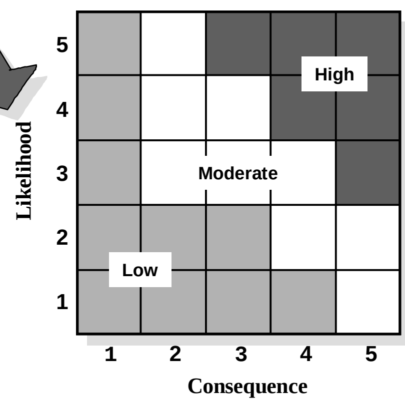


Figure 8-1. Risk Probability and Consequence

In applying this figure, many programs use colors, such as Green = Low Risk, Yellow = Moderate Risk, and Red = High Risk. The derived risk assessment level should correlate with the assessment made by experienced program office personnel; if not, the levels of process variance/likelihood and consequence should be reevaluated and the risk assessment reconsidered.

In practice, likelihood of occurrence is usually a judgment call whereas process variance may be somewhat easier to measure although sound judgment will still be required. Knowing this underscores the importance of experienced and/or expert judgment to truly assess program risk. Chapter 5, "Practice Engineering Fundamentals" should be consulted to gain insight into technical baseline Best Practices and "Watch Out Fors," related to critical processes in design, test, and production and the principal areas of risk associated with each of those processes.

Risk Analysis and Mitigation

Once risk has been identified and assessed, the next step requires Risk Analysis and Mitigation. As part of this step, the risk owner develops specific tasks that, when implemented, will reduce the stated risk to an acceptable level. This does not necessarily mean reducing the risk to low. Some programs consider "no risk" as "no progress" and encourage proactive pursuit of cutting edge technologies. This may require accepting some level of risk if the result leads to future gains in terms of performance, schedule and/or cost.

The risk analysis process requires localizing the source or cause of the identified risk, being careful not to confuse symptoms with cause. It is the source/cause which will receive the necessary resources to mitigate risk to an acceptable level. Once this has been accomplished, Mitigation Plans must be developed that describe what has to be done, when, by whom, the level of effort, and the material or facilities required to mitigate risk to an acceptable level. A proposed schedule for accomplishing these actions is required as well as a cost estimate if possible. All assumptions used in the development of the Mitigation Plan must be listed. Recommended mitigation actions that require resources outside the scope of a contract, Ship Project Directive, Work Request, or other official tasking should be clearly identified. The risk form used by the program should also include a list of the IPT(s), which the risk area or the Mitigation Plan may impact. When completed and approved by the cognizant individual, the risk form is recorded/entered into the database.

Figure 1-1, Chapter 1, lists some ideas for developing risk Mitigation Plans that are self-explanatory. Two items listed in the ToolBox also contain a couple of sources that may benefit Mitigation Plan development. These include, but are not limited to the "DoD 4245.7-M Templates" and "NAVSO P-6071 Best Practices" manuals. These documents are often useful in developing Mitigation Plans for Design, Test, and Manufacturing risk areas. The idea "Renegotiate Requirements" should normally be recommended as a last resort.

Another consideration is the identification of interrelationships between identified critical risks and risk mitigation plans. For example, in developing risk Mitigation Plans, a common Mitigation Plan could be used to mitigate several areas of risk (e.g., improved convection cooling techniques could reduce system complexity by eliminating the need for forced air cooling and improving part design margins by reducing worst case operating temperatures). Conversely, plans developed for mitigating risks in one area could have an adverse effect on other risks (e.g., the addition of heat sinks to improve convection cooling could adversely increase system weight and increase maintenance times). This type of analysis is encouraged to ensure that a Mitigation Plan for one area of risk does not have a counter-productive effect on one or more other risk areas.

Do not expect to avoid risk completely; every program, be it an ACAT I or ACAT IV, will have risks. Once risks have been reported and assessed, a mitigation strategy for every moderate and high risk should be established. Risk resolution and workarounds can be kept off the critical path by early identification and resolution. The program office has, as a minimum, three risk mitigation strategies available: risk reduction/prevention, risk transfer (sharing) and risk acceptance.

PM Note

- **Risk Reduction/Prevention**: Mitigation actions should clearly identify the root cause of the risk, how the root cause will be eliminated/reduced, and who (individuals or teams) are responsible for carrying out these actions. Progress against mitigation actions must be tracked at appropriate intervals. While this is often done at milestone reviews and other major program decision points, it is in the best interest of the program to review these efforts continuously. One way to accomplish this is through the use of Event Driven Risk Mitigation Plans (discussed under "Reporting the Risk"), in which risk mitigation activities are integrated with the overall program schedule and resources.

- **Risk Transfer (or Sharing)**: In some cases, risk consequence must be shared with another party, such as the contractor or a participating program office. Risk can also be transferred or reallocated to different WBS elements or subsystems. In this instance, reallocation is appropriate only if the element to which it is reallocated is better suited to mitigate the risk. Risk transfer may be appropriate when the consequence of risk is high but the likelihood of occurrence is low. Transfer techniques, for example, can include warranties or insurance policies.

- **Risk Acceptance**: As stated previously, every program has risk. Generally, the more the program pushes state-of-the-art technology, and the greater the performance and operational requirements, the greater the risks. In many cases, the program manager must be willing to accept some of these inherent risks, as reduced risk would come at the expense of a degraded mission and performance, and adversely impact budget and schedule constraints. The key in accepting these risks is that the program manager must ensure that these risks are identified and understood early so that they do not become problems later and adversely impact the program.

Tracking the Risk

As part of the assessment and reporting processes, program risks must be formally tracked and documented in an organized manner. This is necessary to determine trends and keep a status on risks and the effectiveness of mitigation activities. Individuals report data in different ways; therefore, it is imperative that all members of the risk team, which includes the contractor:

- Use a standard reporting format, and
- Use the same terminology and definitions to describe, define, and report risk

A standard format allows data to be communicated effectively between team members and management, and allows standardized data to be incorporated into a risk database. A sample Risk Assessment Form (RAF) is shown in figure 8-2.

An effective tracking system has the following characteristics:

- Risk data, decisions, and mitigation activities are accessible to all team members and program office personnel involved with risk management, and
- Risk data is compiled in a central database, so that data can be retrieved and put into useful formats for analysis and reporting

Database Software

“Manual” databases will accomplish the job of tracking, however, electronic databases are preferred because they offer access at remote locations, access by several personnel at once, rapid recall and sorting of data, and links to contractor risk databases. Several recommended database programs, available as COTS items, include Lotus Notes®, Microsoft FOXPRO®, and Microsoft Access®. The database should, as a minimum, include the fields contained on the RAF. Creating the database can be as simple as making the RAF, or a modified version thereof, the opening menu of the database with each field in the form being a drop down menu. The New Attack Submarine On Line Risk Database (OLRDB) was developed by the Program Executive Officer (Submarines) (PEO(SUB)) to identify, assess, manage, track and report program risk. This is a Government owned tool, and information on obtaining an electronic copy of the OLRDB shell will be provided on the ASN(RD&A)ABM homepage:

Dial up the ABM homepage for free software information

http://www.abm.rda.hq.navy.mil

When choosing a database format, consider the following:

- Coordinate with the prime contractor on the database program you/he will be using. Use of the same/compatible software will ensure unhindered data flow, access, and sharing of information between the program office and contractor
- Include database requirements in the contract or Statement of Work

- Use COTS software if possible; this will ensure that software packages and upgrades will be available to the prime contractor and any new contractor/supplier that will need to access the system
- On-line systems allow remote access between all parties. Databases should be secure to prevent unauthorized access

Reporting the Risk

A 1998 GAO Report (GAO/NSIAD-98-56), "Best Practices, Successful Application to Weapon Acquisitions Requires Changes in DoD's Environment," noted that industry encourages and rewards personnel who report risk; whereas in the DoD, "*problems or indications that the [technology, cost and schedule] estimates are decaying do not help sustain the program in subsequent years, and thus their admission is discouraged.*" It further stated that there were "*few rewards for discovering and recognizing problems early in DoD program development, given the amount of external scrutiny the programs receive.*" A 1994 study by the Defense Systems Management College also reported that "*A feeling of responsibility for program advocacy appears to be the primary factor causing Government managers to search aggressively and optimistically for good news relating to their programs, and to avoid bad news, even when it means discrediting conventional management tools that forecast significant negative deviations from plan.*"

Encourage your staff to report risk

The above findings reflect a culture problem within DoD that requires change. Since program risks are unknowns, at least until they have been assessed, all risks are inherently high or "Red", until their impact is further understood and/or mitigated. In order to mitigate program risk, risks must be reported. To ensure all risks are reported and not understated, program managers should employ the following:

- Strongly encourage the reporting of risk without fear of reprisal
- Status all new risks as high or "Red" until consequences and program impact are understood and/or mitigated

Use standard reporting formats

Risks must be presented in a clear, concise, and standardized format so that senior personnel responsible for making programmatic and technical decisions are not burdened with large amounts of non-standardized data, yet at the same time have enough information to make programmatic decisions. The report format should be comprehensive enough to give an overall assessment, be supported by enough technical detail and include recommended corrective actions. The Under Secretary of Defense for Acquisition and Technology (USD(A&T)) designates certain ACAT I programs to submit a Defense Acquisition Executive Summary (DAES) report. The purpose of this report is to highlight risks and actual problems to USD(A&T) before they become major problems. For designated ACAT I programs, high and moderate risks should be included in the DAES report. The Program Risk Mitigation Waterfall Chart, Figure 8-3, illustrates the connection between program events and mitigation efforts as well as a record of progress in risk mitigation. In addition, the following provides some basics in reporting or "rolling up" risk data for senior personnel in a position to assist the program in reducing risk:

- Use of a standard form and format to report risks
- The risk database should be capable of generating risk reports
- Government and contractor reports should utilize the same form/format and terminology
- Reports should summarize high and moderate risks and recommended actions
- A "watch list," which tracks risks and mitigation activities should be included as part of the report. The watch list is normally updated monthly and should include the top ten (or more as applicable) risk items prioritized by exposure and leverage
- Reports should also include, as a minimum:
  - Number of risk items resolved to date
  - Number of new risk items since last report
  - Number of unresolved risk items
  - Unresolved risk items on the critical path
  - Effect of technical risk on cost and schedule

RISK ASSESSMENT FORM

Please fill out and submit to RM Coordinator. Use additional pages if needed.

- email :
- Fax # :

RISK TRACKING NUMBER
(Assigned by RM Coordinator.)
XXX-XXX-XXX

RISK TITLE:

OVERALL RISK LEVEL
(See reverse. Circle one.)
LOW MODERATE HIGH

PRODUCT / SUBASSEMBLY / CONFIGURATION ITEM:

PROCESS AREA OR TEMPLATE
(To be identified by RM Coordinator.)

RISK LEVEL IDENTIFIERS
(Enter a number 1-5. See reverse.)
Process Variance: ____
Consequence:

- Performance ____
- Schedule ____
- Cost ____

REQUIREMENT AFFECTED
(Record par. # and security classification. Provide summary of requirement if unclassified.)
A-Spec _______ PIDS _______ WBS _______ Other ______

DATE
Identified:
Submitted:

RISK ORIGINATOR
Name:
Phone #: IPT:

RISK OWNER
(Assigned by the Risk Originator/IPT)
Name:
Phone #: IPT:

RISK DESCRIPTION: (Provide as much detail as possible. Use IF-THEN-IN ADDITION format.)

RISK LEVEL RATIONALE: (Use Risk Level Identifiers (see reverse) and Program Risk Level Standard Guidelines.)

RISK MITIGATION RECOMMENDATIONS:
(What actions will be/have been taken to mitigate this Risk and when? How could have we avoided this Risk?)

Figure 8-2. Sample Risk Assessment Form

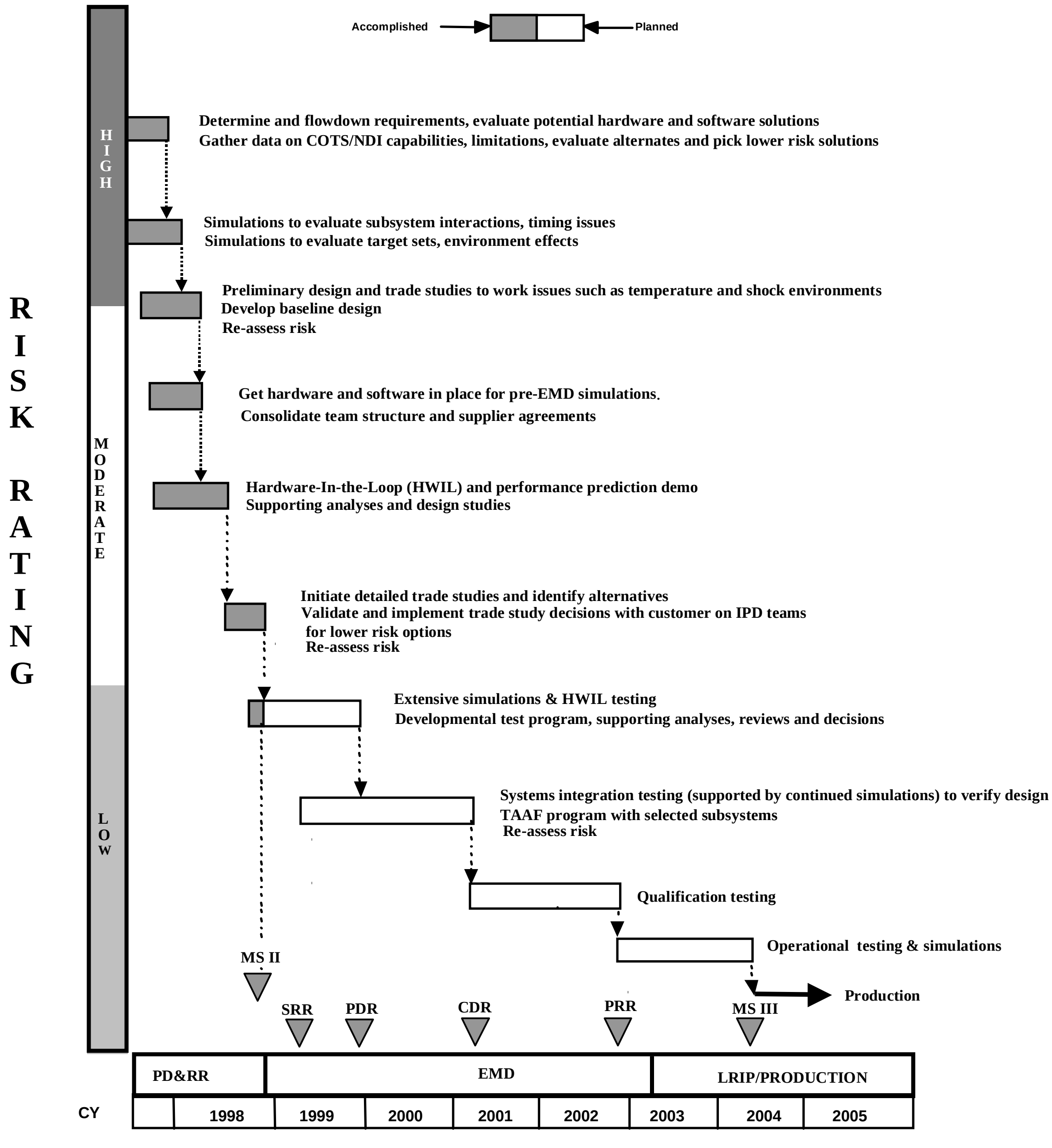


Figure 8-3. Program Risk Mitigation Waterfall Chart (Example)

Figure 8-4 is a Sample Watch List based on the WBS approach that rolls up technical risk, schedule and cost against the Allowable Unit Production Cost (AUPC) associated with program risk, with cost impact quantified for each risk. The expected cost is computed by adding the specific risk mitigation cost to the program schedule cost (schedule slip times program burn rate) and multiplying the sum by the probability of the risk occurring. Expected costs of individual risks are totaled to provide an expected program risk cost.

		Risk Assessment				
Risk #	Specific Area	Technical	Schedule	Cost	AUPC	Expected Risk Cost $M
1	Software Productivity	High	Med	High	Med	3.5
2	Seeker Development	High	High	Med	Med	3.0
3	Survivability	High	High	Med	Med	2.7
4	Antenna Development	Med	Med	Med	Low	1.3
5	Seeker Supplier Availability	Med	Med	Med	Low	2.0
6	Seeker Target Acquisition	Med	Low	Med	Low	0.7
7	Missile Vibration	Med	Med	Low	Low	0.5

Figure 8-4. Sample Watch List

Chapter 9
Use Independent Assessors

What is the Relationship Between "*Independent Assessors*" and Technical Risk?

A disciplined approach to technical risk management can be extremely challenging in today's defense acquisition environment. Not only are Government and industry organizations experiencing significant downsizing, but also many of those leaving their jobs are the most experienced personnel. As a result of Acquisition Reform and revised DoD 5000 series documents, the policies and procedures governing the procurement of military hardware and software have undergone major changes, e.g., the transition to commercial practices and the emphasis on Commercial-Off-the-Shelf/Non Developmental Items. The loss of experienced people, coupled with increasing procedural and technical complexity, means greater risk for Program Managers (PMs). One proven way of reducing this risk is to utilize a team of experienced people to conduct independent assessments of program health.

Few would argue that obtaining a second opinion constitutes a common sense approach prior to making a critical decision regarding important matters, such as major surgery, career changes or financial investments. In defense acquisition, PMs are faced with many critical decisions, and second opinions (e.g., independent assessments) play a key advisory role when those decisions are made, including those pertaining to program technical and management risk.

DoDD 5000.1 requirement for independent assessments

DoDD 5000.1 states that "*Assessments, independent of the developer and the user, are extremely important to ensure an impartial evaluation of program status. Consistent with statutory requirements and good management practice, DoD shall use independent assessments of program status. Senior acquisition officials shall consider these assessments when making decisions. Staff offices that provide independent assessments shall support the orderly progression of programs through the acquisition process. Independent assessments shall be shared with the Integrated Product Team so that there is a full and open discussion of issues with no secrets.*"

Although the first sentence of this quote implies assessments directed by the Director, Operational Test and Evaluation, the remainder of the quote contains generic direction on assessments, which is applicable to all aspects of a program, including risk.

Program Experience

In recent years, several Navy programs have benefited noticeably from recommendations provided by independent assessors. These programs include F/A-18 Aircraft, Consolidated Automated Support System (CASS), New Attack Submarine (NSSN) and Surface Combatant Twenty First Century (SC-21). The scheduling of the assessments varied based on each program's needs at a particular time. NSSN and SC-21 scheduled independent assessments prior to major milestone reviews; CASS scheduled an independent technical review to address poor initial Operational Test and Evaluation results and the need to improve design and manufacturing processes; and F/A-18 uses an independent assessment team on a continuous basis. Irrespective of the timing, independent assessments have proven to be a valuable tool for a better understanding of Navy program risks. Outside of DoD, NASA is a strong proponent of independent and timely technical reviews, e.g., reviews of analyses, many highly specialized, pertaining to the reliability/design process. NASA Practice No. PD-AP-1302 notes that approximately 40 percent of all analyses contain significant shortcomings when performed for the first time. Roughly half of these are defects or omissions in the analysis itself and not design defects. The other 20 percent represent design defects, the severity of which varies from minor to mission catastrophic. *The only proven method for detection of these defects is an independent review of the design details by an impartial, objective, competent peer group in the appropriate technical field.*

Tasks

In preparing for an independent risk assessment of a specific program, the tasks assigned to the assessors should include, but not be limited to, the following:

- Review the program's risk management approach and the status of risk assessments conducted to date
- Review Mission Needs Statement directed actions, Cost and Operational Effectiveness Analysis results, Acquisition Strategy, Operational Requirement Document and other relevant program documentation (e.g., Design Reference Mission Profile) for any known or potential risk areas/critical processes which may have been overlooked
- Examine all advanced or emerging technologies being considered and determine any known or potential risk areas/critical processes which may have been overlooked

- Prepare a final report for the PM concerning the adequacy of risk management efforts to date, readiness for the next milestone review, and recommendations for improvement

Final thoughts

In summary, an independent risk assessment is a high-payoff tool for the PM's use in determining the adequacy of his risk management process. Assessors should be independent of the PM's staff and selected on the basis of their professional reputation, their in-depth experience, and their willingness to serve as "honest brokers" in behalf of the program being reviewed.

Chapter 10

Stay Current on Risk Management Initiatives

What is the Relationship Between "*Risk Management Initiatives*" and Technical Risk?

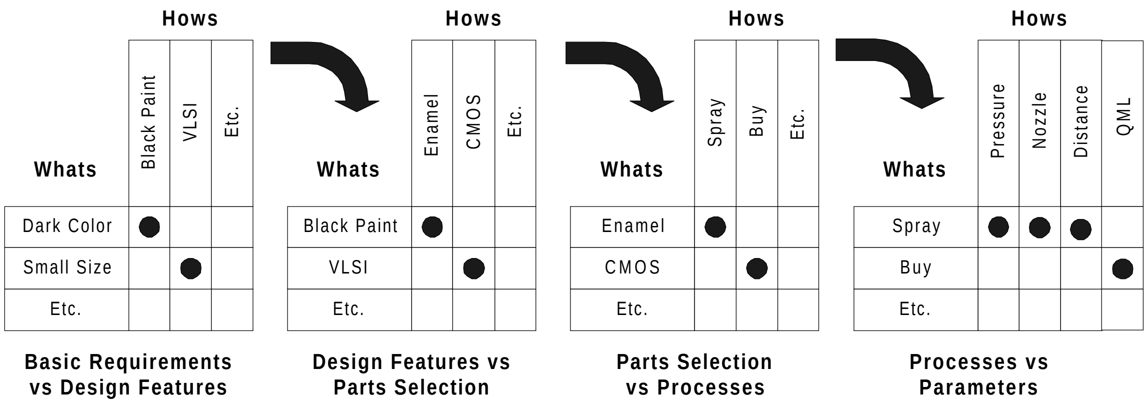
The Program Manager (PM) and staff often are not aware and consequently do not take advantage of continual advancements and new initiatives in best practices and analytical tools. Additionally, state-of-the-art expertise, such as that available from ManTech Program Centers of Excellence, the DoD Information Analysis Centers, and Government Industry Data Exchange Program (GIDEP) can provide valuable lessons learned to reduce technical risks. Awareness of continual advancements and new initiatives in best practices and analytical tools provides opportunities for more effective development and manufacture of products meeting customer requirements with less technical risk. These initiatives generally have focused on a better understanding of customer requirements, improvements in the design and manufacturing processes, and methods for reducing variation in the product and related processes. These new initiatives enhance the achievement of robust designs and aid in further reducing the variations that occur in products, their performance and associated processes. The following are brief descriptions of several initiatives that enable significant technical risk avoidance.

Quality Function Deployment

☜ ***What is QFD?***

The Quality Function Deployment (QFD) is an analysis technique that enables the identification and systematic translation of customer requirements to actions required by the contractor to meet the customer's desires. This technique is based on the use of a matrix to compare *what* the customer wants to alternative ways of *how* the contractor plans to provide it, thereby reducing technical risk.

Although a QFD analysis may be adequate with the use of only a top-level matrix, the cascading of the matrix to lower indenture levels as the design progresses is necessary to identify critical process parameters that must be controlled to meet customer requirements. This cascading also provides a trail from customer requirements to the process parameters that need to be controlled. Identification of the critical requirements and processes, from among many, together with the required controls, are key ingredients for focusing the technical risk management effort. A key to the successful use of this tool is its integration of design considerations and process activities. Shorter development-to-production time with fewer engineering and manufacturing changes are benefits resulting from the use of this tool to manage technical risk.



Source: Reliability Analysis Center START Publication, “Quality Functional Deployment,” Volume 4, Number 1

Figure 10-1. QFD Technique

The following cascading matrixes shown in Figure 10-1, conceptually illustrate one simplified use of the QFD technique:

- The first matrix matched the customer’s requirements (the *whats* identified in rows) to the design features (the *hows* identified in columns) intended to meet the requirements
- The *hows* become the *whats (*design features) of the second matrix as against *hows* (parts selected to implement the *whats*)
- The parts selected then become the *whats* (parts selection) of the third matrix, plotted against the *hows* of the processes used to create the parts
- Finally, the processes become the *whats (processes)* of the fourth matrix, where the *hows* are the process parameters that must be controlled. Thus, the cascaded matrixes translate the customer’s requirements to a set of process parameters to be controlled. One such translation in the last matrix relates the customer’s requirement for a dark color to the pressure of a spray paint nozzle

Investment of time to perform this analysis early in the development program results in insight into the customer's requirements, an understanding of critical process parameters, and a shorter overall product development cycle accomplished at lower risk.

Taguchi Techniques

☞ ***What are Taguchi Techniques?***

The Taguchi techniques are often used to reduce variation in critical areas identified through QFD analysis. The Taguchi techniques are innovative approaches to the statistical design of experiments focused on reducing variation from a targeted value, not from specification limits. It focuses on:

- Identifying the critical factors, both controllable and not, which affect a process or product
- Reducing its sensitivity to variations from various sources, thereby improving quality at optimum cost

After identifying the ideal functions or characteristics of a product or process, team brainstorming is used to identify all possible factors that may affect it, and to select the most important ones to analyze or test. These characteristics and factors are included in a Taguchi orthogonal array to determine optimum solutions to improve quality or reduce variation. The values of these factors or parameters are varied for all characteristics while observing the deviation from the desired target. The resulting statistical information allows the development of a robust product or process that meets the customer requirements, is produced at a lower risk, and is reproducible at the lowest cost.

The Taguchi Quality Loss Function (QLF) provides an approximation of monetary loss caused when a product or process function or characteristic deviates from its targeted value. Deviation from the targeted value results in decrease in quality, customer dissatisfaction and increased loss. The loss can be defined in a broad manner and may include the "hidden factory," performance deficiencies, timeliness, cost increases, customer complaints, warranty costs, market share, reputation, etc. (Ref. ASI Press, "Taguchi Methods and Quality Function Deployment," 1998.)

Technical Performance Measurement

☞ ***What is TPM?***

Technical Performance Measurement (TPM) is, simply speaking, a time-phased progress plan for the achievement of critical Technical Performance Parameters (TPPs). TPPs selected for inclusion should indicate, when achieved, progress in key areas of technical risk reduction and expected program success. TPPs can be related to hardware, software, human factors, logistics or any product or functional area of a system.

TPM helps the PM remain focused on the critical technical elements of a system or program since decisions made more knowledgeably and quickly in these key areas keep a program on track for successful completion. Figure 10-2 illustrates the methodology used to establish a TPM technical baseline and track progress against that baseline.

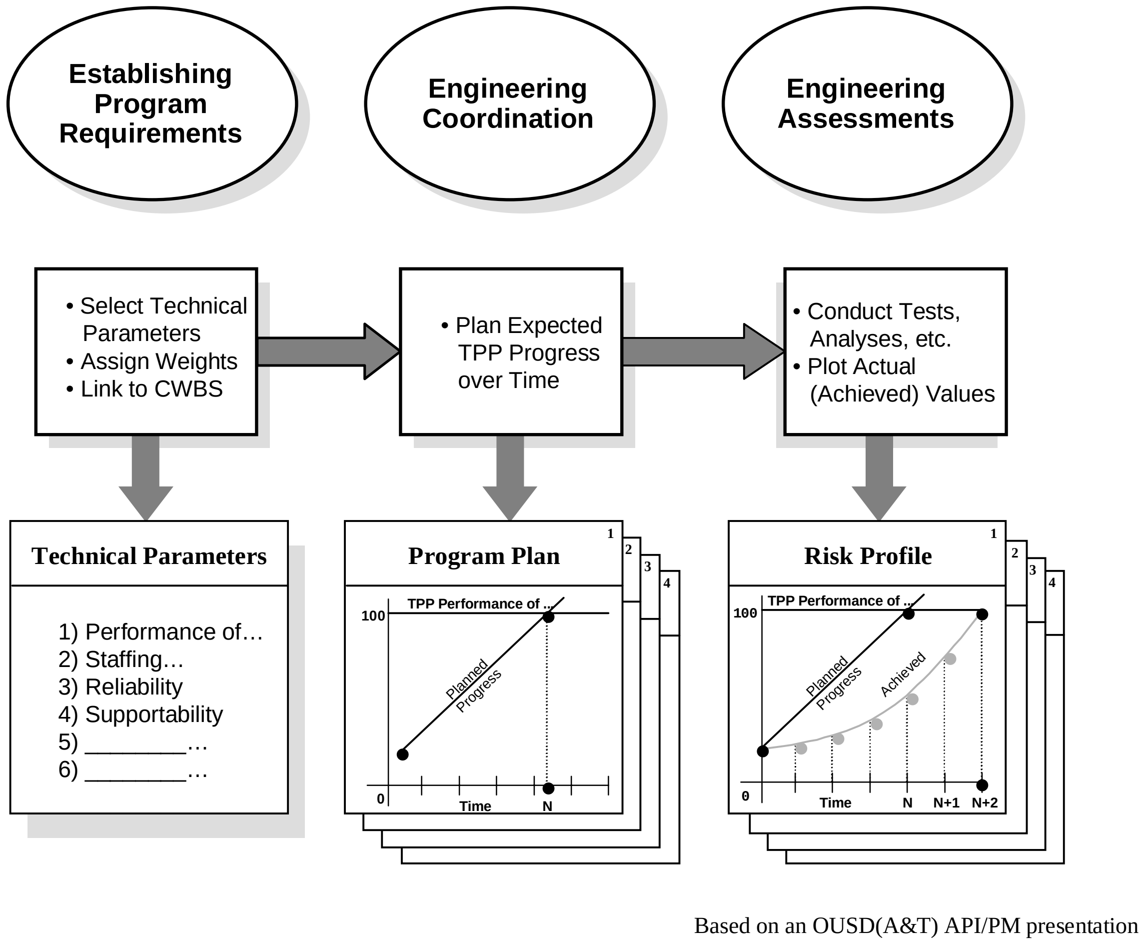


Figure 10-2. TPM Methodology Overview

Properly established and implemented, TPM facilitates identification and response to system/program risks by comparing actual performance to planned TPPs, evaluating significant variances, and instituting corrective actions as needed

More specifically:

- Achieved values (actual test or analytical results) are compared to the progress plan's TPPs to identify variances
- Variances can indicate the level of risk associated with particular processes or elements, depending on degree

- Program success or failure can be estimated/projected by considering the combined effect of risk associated with multiple achieved values
- Corrective actions should be implemented based on assessed levels of risk
- Achieved values should be repeatedly calculated in order to track the success of corrective actions
- Repeated calculations of achieved values also permit the detection of new risks before their effects on cost and/or schedule are irrevocable
- Achieved values that meet TPPs indicate an effective risk-handling strategy and an on-track program

In figure 10-3, the horizontal line at 32.5 lbs. is the planned, final weight for the component — the TPP. Sloping line A indicates the actual/ achieved progress toward meeting this specific parameter, while sloping line B depicts the expected progress in weight reduction. The variance (shaded area) between "actual" and "expected" represents the degree of risk, while progress is being achieved.

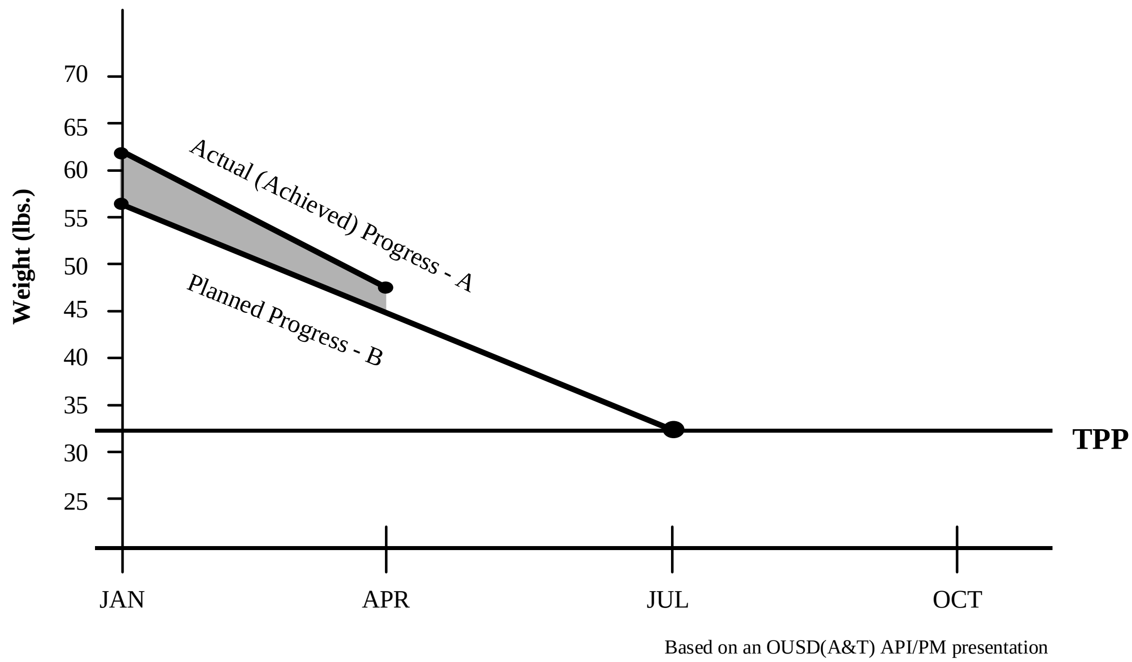


Figure 10-3. Example TPM Progress Chart

TPM System (TPMS) software is available to help PMs by automating the tasks associated with TPM. The TPMS software is a *Government-owned* tool sponsored by the Office of the Undersecretary of Defense (Acquisition & Technology) Acquisition Program Integration & Performance Management. TPMS is free to all Defense Departments and Agencies. Additional information is available at:

http://www.acq.osd.mil/api/tpm/

Free software and info. available at...

Earned Value Management

☜

What is Earned Value Management?

Earned Value Management is a technique that relates resource planning to schedules and to technical performance requirements. All work is planned, budgeted, and scheduled in time-phased "planned value" increments constituting a cost and schedule measurement baseline. There are two major objectives of an earned value system:

- encourage contractors to use effective internal cost and schedule management control system
- provide the customer timely data produced by those systems for determining contract status.

Earned value management is useful in monitoring the effectiveness of risk-handling actions in that it provides periodic comparisons of the actual work accomplished in terms of cost and schedule with the work planned and budgeted. These comparisons are made using a performance baseline that is established by the contractor and the PM at the beginning of the contract period. This is accomplished through the Integrated Baseline Review process. The baseline must capture the entire technical scope of the program in detailed work packages, and includes the schedule to meet the requirements and the resources to be applied to each work package. Specific risk-handling actions should be included in these packages.

The periodic earned value data can provide indications of risk and the effectiveness of risk-handling actions. When variances in cost or schedule begin to appear in the work packages containing risk-handling actions, the appropriate Integrated Product Teams can analyze the data to isolate the causes of the variances and gain insights into the need to modify risk-handling actions.

The benefits to project management of the earned value approach come from the disciplined planning conducted, and the availability of metrics which show real variances from plan in order to generate necessary corrective actions.

Additional guidance...

Detailed implementation guidance may be found in the "*Earned Value Management Implementation Guide*" NAVSO PAMPHLET 3627, Revision 1, of 3 Oct 97.

Chapter 11
Evaluate New Acquisition Policies

What is the Relationship Between "*Changes in Acquisition Policies*" and Technical Risk?

DoD establishes rules and regulations that apply to all DoD agencies and programs. These requirements are often mandated by public law, which the DoD is required to implement. Others are instituted in an effort to bring efficiencies into the DoD. Practices and policies that have been in place for years are now superseded or have been significantly changed. Therefore, it is critical that someone in the program office is assigned the responsibility to not only become familiar with each new acquisition policy, but also understand how it will impact technical risk. Without this understanding, the risks to the program may be high.

Cost As An Independent Variable

☜ ***What is CAIV?***

In December of 1995 the Undersecretary of Defense (Acquisition and Technology) introduced a new concept entitled "Cost as an Independent Variable" or CAIV. The intent of CAIV is to provide the customer/warfighter with highly capable systems that are affordable over their life cycles. CAIV is based on the principle that the best time to reduce Total Ownership Cost (TOC) is early in the acquisition process and that initial cost-performance trade-off analyses should be conducted before the operational requirements and acquisition approach are finalized.

CAIV and Risk Management

CAIV requires that Program Managers (PMs) establish aggressive cost objectives. The ability to set and achieve such cost objectives depends significantly on early trade-offs in performance versus cost. The maximum level of acceptable risk is one of the factors that help to define an aggressive cost objective. Risks in achieving both performance and aggressive cost goals must be clearly recognized and actively managed through continuing iterations of cost/performance/schedule/risk tradeoffs, identifying key design, test and manufacturing process uncertainties and developing and implementing solutions.

Examples of such solutions include:

- Aggressive cost reduction measures, such as the elimination of design analyses, reduction in trade studies, or the elimination of subsystem testing, may significantly increase the risk of meeting performance and schedule thresholds
- Driving the design to achieve maximum performance may significantly increase the risk of meeting cost and schedule thresholds, perhaps with little operational need for the extra performance

User participation in trade-off analyses is essential to attain a favorable balance between performance, cost, schedule and risk. The PM and user representatives should identify risk and cost driving requirements during the generation of the Operational Requirements Document (ORD), in order to know where trade-offs may be necessary. Integrated Product Teams (IPTs), especially during trade studies, should address best practices and their impact on program cost and schedule risks. The approval and funding of risk-handling options should be part of the process that establishes CAIV cost and performance goals. Improving risk management will enable PMs to support the CAIV concept of setting early cost objectives that are challenging but realistic. Program planning and Integrated Baseline Reviews should be conducted with an understanding of the scope of technical work required to manage program risks. TOC objectives should be developed and included in the ORD, solicitations and contracts. Several Best Practices and “Watch-Out-Fors” are shown in Table 11-1.

Table 11-1. CAIV Best Practices and Watch-Out-Fors

Best Practice

- An aggressive and structured risk management program is implemented to manage trades between performance, cost, and schedule
- Out-year resources identified
- Production and Operational and Support (O&S) cost objectives included in the Request for Proposal (RFP)
- Incentives for achieving cost objectives included in RFP and contract (% relative to total contract dollars)
- Mechanism for contractor suggestions to reduce production and O&S costs in place and operating
- Allocation of cost objectives provided to IPTs and key suppliers
- Identification and implementation of new technologies and manufacturing processes that can reduce costs
- Identification of procedural/process impediments to cost reduction measures
- Establishment of a strong relationship with vendor base, including sound incentives structure

Watch Out For

- Cost objectives not defined or consistent with program requirements and projected fiscal resources
- No Government or contractor management commitment to achieve cost objectives
- Technical IPT members not participating in defining alternative methods of achieving requirements
- “Watch-Out-Fors” not addressed and evaluated to achieve acceptable program technical risk
- Design-To-Unit-Cost (DTUC) in EMD that does not consider trade-off with various levels or types of warranties that may arise during production award negotiations. Negotiations may result in increased warranty costs that exceed the program’s planned allocations. Additional design efforts during EMD could mitigate this program risk
- Source selection, DTUC, and cost of ownership, based in part on a true “bumper-to-bumper” warranty during production, when a limited maintenance type contract may actually be the more likely outcome depending on the acquisition strategy
- Expectations of contracting for a “bumper-to-bumper” warranty when program funding is not adequate

Additional information may be obtained from the following:

Additional on-line information

OSD: http://www.acq.osd.mil/log/lro/toolkit/opening.html
Navy: http://www.acq-ref.navy.mil/turbo/14.htm

Environmental Risk Management

What goes into a weapon system, facility, or platform must eventually come out – either during use, during refurbishment, at system retirement, or as a by-product. With this in mind, the goal of this section is to make PMs aware of the latest technologies and resources available to help bring and/or retain acquisition programs in compliance with Federal environmental mandates.

☜ ***Focus For the PM***

PMs are responsible for understanding real and potential negative environmental effects caused by their programs throughout the entire life cycle of the system. They must eliminate, minimize, and mitigate as many of these effects as possible, which may increase risk to the program.

One of the most effective ways to minimize environmental risk and associated costs during the life cycle of a system is through Pollution Prevention, or P2.

Available Environmental Information

There is no single resource list of the most environmentally sound designs or processes for all situations. There are, however, quite a few joint Government/Industry efforts through which the PM can access the latest, validated P2 Engineering Technologies for a great number of situations and uses. For instance:

Toxics Use Reduction Institute (TURI) —
- Conducts technical research on reduction technologies and processes associated with the use of specific toxic substances
- Provides data on the technical feasibility of reduction technologies and processes associated with the use of specific toxic substances
- Provides technical support to and/or facilitation of the transfer of these technologies to industry

P2 GEMS Internet Search Tool (managed by TURI) —
- Contains management information about environmental technical concerns, environmental issues, and materials friendly to the environment for facility planners, engineers, and managers.
- Allows search by keyword or by selection from one of four categories: 1) Product or Industry, 2) Chemical or Waste, 3) Management Tools, or 4) Processes

Pacific Northwest Pollution Prevention Resource Center (PPRC) —
- A non-profit organization supporting projects that yield results for pollution prevention or for the reduction/elimination of the use of toxic substances
- Maintains a series of tools, technical publications, and assistance networks including the PPRC Research Projects Data Base, which provides technical

information on 1) pollution prevention, 2) toxic material alternatives, 3) application, methods and processes, and 4) energy efficient technologies

Enviroene Database (Maintained as part of EPA's homepage) —
- Contains extensive pollution prevention/compliance assurance data
- Includes pollution prevention case studies, technologies, and information on alternatives to toxic solvents
- Contains 1) a list of P2 programs nationwide, 2) P2 technical/research development data, and 3) the Solvent Substitution Data Base System (SSDS) Umbrella with additional P2 data bases focused on specific environmental applications

Enviroene DoN P2 Programs Database (Maintained by the Navy) —
- Provides on-line access to Navy efforts and best practices aimed at reducing the use of hazardous materials in existing operations/processes and at preventing the production of polluting agents
- Includes model P2 plans, case studies, fact sheets, and other helpful material

Solvent Handbook Database System (SHDS – Maintained by DoE & DoD) —
- Identifies alternative solvents (i.e., solvents currently restricted) and evaluates their performance, corrosive potential, air emissions, recycling capabilities, and compatibility with other materials

Joint Group for Acquisition P2 Projects (JGAPP) —
- Conducts technical research on Industry/Government P2 programs aimed at eliminating regulated materials from selected weapons systems, for example, "Alternatives to Tin-Lead Surface Finish on Circuit Boards" and "Non-Chrome Primers"

ABM Environmental Homepage (Maintained by ASN(RD&A)ABM) —
- Links directly to environmental requirements lists, environmental responsibilities lists, environmental contract clauses, and DoD Memoranda/Executive Orders related to environmental issues

How Can PMs Get This Information?

All the above tools and organizations can be reached via the World Wide Web and can be accessed through links on the ASN(RD&A)ABM Homepage.

Dial up the ABM Homepage for quick results!

http://www.abm.rda.hq.navy.mil

Single Process Initiative

The Single Process Initiative (SPI) is a DoD acquisition reform initiative introduced in December 1995, as a means for contractors to replace multiple, Government-unique, business management and manufacturing practices with facility-wide and more recently, corporate-wide practices. The goal of SPI is to reduce contract costs associated with unnecessary Government requirements, and to move towards common acquisition practices within DoD and industry. SPIs are established based on expected return on investment, quality improvements, and/or strategic importance.

☞

What is SPI?

Single Process Initiative and Risk Management

When Industry submits a proposal to replace or eliminate previously approved and successful military processes with commercial or company processes, technical risks may exist until the new processes have been proven satisfactory. To minimize these technical risks, the following guidance in Table 11-2 should be considered.

Table 11-2. SPI Best Practices and Watch-Out-Fors

Best Practice

- Joint Government and contractor Management Council teams consider:
  - The unique needs and risks of each program
  - The benefits of common processes
  - The robustness of the contractor's process controls
- Prime contractor flow-down requirements to subcontractors based on accepted industry standards or acceptable subcontractor common processes vice unique in-house standards
- Prime contractor review subcontractor SPI processes to assess adequacy or risk in meeting program objectives or requirements
- Contractor demonstrates process control on proposed common processes before the proposed process is implemented
- Government and the contractor need to precisely define the facility location(s) at which the contractor's proposed single processes apply

Watch Out For

- Potential cost and schedule impact of establishing common depot repair processes between competitive depots on tooling, test equipment, fixtures, and capitalized equipment
- Contractor SPI processes that are not technically acceptable for meeting unique program requirements
- A Management Council that does not keep the IPTs informed about the overall Management Council strategy and status

The Assistant Secretary of the Navy (Research, Development and & Acquisition) Acquisition Reform Office is the Navy's lead office for SPI activities. For the latest information on approved SPIs, SPI status, etc., access the Navy and DCMC SPI databases at:

Navy: http://www.acq-ref.navy.mil/spi.html
DCMC: http://www.dcmc.hq.dla.mil/spi/spi.htm

Latest SPI information available on-line!

Diminishing Manufacturing Sources & Material Shortages

Numerous parts vital to repairing and supporting older equipments are being discontinued by manufacturers every year. During fiscal year 1996, the Government Industry Data Exchange Program (GIDEP) alone distributed 125 discontinuance notices affecting 50,600 parts. Systems designed for long operational periods are being supported by an industry where parts changeover and obsolescence are measured in months. Once a system is out of initial production, the contractor or parts manufacturers discontinue the parts or product lines, an issue that has become even more critical due to the reduced amount of Government insight under acquisition reform. This issue is referred to as Diminishing Manufacturing Sources and Material Shortages (DMSMS). Following are some DoD and industry resources and practices that the PM can use to reduce program risk relating to DMSMS.

What is DMSMS?

DoD Resources. The following is a partial listing of DoD organizations supporting DMSMS efforts:

- **Government Industry Data Exchange Program**. GIDEP is the centralized DoD repository for DMSMS information (cases) and solutions. GIDEP maintains notifications from manufacturers, original equipment manufacturers, and Government activities of items that are no longer being produced. DMSMS Notices are issued for any type of discontinuance, obsolescence or shortage. GIDEP also provides the value added to DMSMS notices by adding information from third party sources. GIDEP DMSMS data includes alternate sources, manufacturer's data, after market suppliers, and can also compare the user's bill of materials or parts list with parts listed in the database. When the user requests these services, a list of affected parts is returned for analysis. GIDEP membership is open to Government agencies and suppliers to the US and Canadian Governments. Information about GIDEP can be obtained from:

  GIDEP Operations Center
  P.O. Box 8000
  Corona, CA 91718-8000
  Web Page URL: http://www.gidep.org

GIDEP provides links to several DoD and Industry DMSMS sites. A sample of these sites is provided below. For a more complete list, access the GIDEP home page and click on the DMSMS icon.

More information available on-line!

- **DoD DMSMS Working Group**: The DoD has established a DMSMS working group to assist program offices in finding DMSMS solutions. This team is comprised of members from the different DoD program offices and activities who have mutual DSMS concerns

- **Defense Microelectronics Activity (DMEA)**: Maintains a centralized source for re-engineering components and establishing manufacturing sources for technology insertion. DMEA provides technologically correct and economically viable solutions for microelectronics obsolescence **(http://www.dmea.osd.mil)**

- **Defense Supply Center, Columbus (DSCC)**: Formerly called DESC, provides DMSMS case information on recent discountenances **(http://www.dscc.dla.mil/programs/dsmms/index.html)**

Industry Resources. The following is a partial listing of private companies that will provide DMSMS solutions to subscribers and members:

- **TACTech, Inc**.: Provides an information service that furnishes users with proprietary information including life cycle projections for microcircuits and discrete semiconductor devices used in military and Government systems. TACTech licenses its proprietary software and databases to Government contractors and Government agencies. The TACTech library contains information on over 100,000 active and obsolete devices, covering virtually all standard devices. Their software identifies the life cycle of the parts and pending obsolescence **(http://www.tactech.com)**

- **Electronic Industries Alliance**: The EIA maintains a database, available on a subscription basis **(http://www.eia.org)**

- **Semiconductor Industries Association (SIA)**: SIA maintains a semiconductor obsolescence database **(http://www.semichips.org)**

Best Practices. There are a number of possible solutions for DMSMS. All have been used with varying success. A combination of some of the following practices can assist the PM in reducing the risk of DMSMS impact to their programs:

- Early in the acquisition cycle, program office personnel work with industry to perform continuous market research on industry trends

- Because the quantities of components used in any DoD system are small compared to industrial and consumer markets, try to select Qualified Manufacturer's List parts and industrial grade parts in high demand by industry
- Make life-of-type buys
- Use after market suppliers
- Use DMEA for technology insertion, which often reduces component count and number of boards
- Utilize open architecture design for new technology insertion
- Maintain the ability to quantify the impact of discontinuance of hardware availability on deployed systems
- Obtain data at a functional level, which is technology independent and complies with standard product descriptions for use in future emulation of parts and assemblies

Table 11-3, provided by Joint Stars Program Office, illustrates a sample cost impact for various alternatives to solve DMS parts problems.

Table 11-3. Alternatives to DMSMS Parts Problems

TYPE	Possible Provider	Non-recurring Engineering $(K)	Cost multiplier per part (x times original cost)
Alternate Source	Aftermarket vendor	None	4
Substitute	Other Service	None	8
Lifetime Buy	DSCC or part manager	None	15
Repackage	Aftermarket	10-50	20
Remanufacture	Aftermarket or DMEA, HTT, DTC, etc.	30-100	30
Emulation	GEM, DMEA, ITD	20-150	100
Board Redesign	DMEA, HTT, DTC, ITD	100-500+	10,000
COTS Insertion	Commercial	1000-10000	Undetermined

DSCC Defense Supply Center, Columbus
DMEA Defense Microelectronics Activity, Sacramento, CA
HTT Hardware Technology Center, Ogden, UT
DTC DMSMS Technology Center, Crane, In
GEM Generalized Emulation of Microcircuits, Sarnoff Research Center, Princeton, NJ
ITD Institute for Technology Development

Configuration Management

Configuration Management (CM) is defined as a process for establishing and maintaining consistency of a product's performance, and its functional and physical attributes with its design and its operational use throughout its life. As affirmed by MIL-HDBK-61, "the intent of CM is to avoid cost and minimize risk. Those who consider the small investment in the CM process a cost-driver may not be considering the compensating benefits of CM and may be under-estimating the cost, schedule and technical risk of an inadequate or delayed CM process."

DoD 5000.2-R states the requirement for:

> *"…a configuration management process to control the system products, process and related documentation. The configuration management effort includes identifying, documenting and verifying the functional and physical characteristics of an item; recording the configuration of an item; and controlling changes to an item and its documentation. It shall provide a complete audit trail of decisions and design modifications."*

CM is a fundamental process that must be applied for long term product success, regardless of which organization, e.g., Government or contractor, monitors its implementation. The CM process encompasses, to some degree, every item of hardware and software down to the lowest bolt, nut and screw, or lowest software unit.

CM begins during development and, properly implemented, will contribute to the preparation of high quality design release drawings. Such drawings should represent a stable design configuration that is suitable for production, installation, maintenance and logistics support necessary to ensure that the Operating Forces receive, and are able to maintain, weapon systems which work when needed. After delivery of equipment to the warfighters, CM continues whenever this equipment is modified or upgraded because any modification or upgrade program is essentially a new development and production effort requiring the same CM process discipline as a "new start" program.

With the implementation of Acquisition Reform (AR), DoD policies regarding the CM process have changed. The responsibility for the CM process is now shared even more between DoD and the contractor, and typically is no longer the sole responsibility of the program manager. In this regard, AR does not diminish the importance of CM; rather it has resulted in a reconsideration of the degree to which the CM process should be controlled by the Government compared to the contractor. Significant authority for configuration control may be delegated to contractors during all phases of the life cycle, depending on such factors as the acquisition strategy, the maintenance concept and the associated Technical Data Package (TDP). However, DoD ultimately is responsible for the performance and configuration of the systems acquired. The procuring Government agency is always the configuration control authority for the top level attributes, as well as for lower level performance and design attributes depending on the aforementioned factors of acquisition strategy, maintenance concept and TDP. This shift in responsibility, along with the move to Commercial-Off-The-Shelf/Non-Developmental Items (COTS/NDIs), has decreased the program manager's involvement with the configuration of his system. This has increased program risks and the need for the program management offices to plan for and understand the CM process, and to ensure supportability and interoperability of military equipment and software. The following provides some of the more significant items to be considered when managing the configuration of a system, under the new AR policies, to reduce program risks.

Documentation: Acquisition reform has made a significant change in the types of configuration documents used to specify configuration items (CIs). DoD now specifies performance requirements and, in most cases, leaves design solutions to the contractor. The types of documentation needed at the system level are determined by the DoD procuring agency, whereas the contractor may be delegated the responsibility to choose the documentation needed below the system level. DoD policy indicates preference for products meeting performance requirements rather than detailed specifications, wherever possible.

Design Solutions: Acquisition Reform and the latest DoD 5000 series have provided contractors the opportunity to prepare a design solution most suitable to meeting the operational requirement. It is important for the DoD program manager to recognize that there will be a great deal of diversity in the methodologies employed by various contractors, and consequently, an early emphasis on CM process discipline will pay dividends in the long run, e.g., by ensuring compatibility, maintainability and supportability at all levels of repair.

Configuration Control Authority: Configuration control is the process used by contractors and program managers to establish the configuration baseline and manage the preparation, justification, evaluation, coordination, disposition, and implementation of proposed engineering changes, and deviations to effected CIs and baseline configuration documentation. The DoD needs to take delivery of and control product configuration documentation at a level of detail commensurate with the operational, support and reprocurement strategies for a given program. For reparable CIs, design disclosure documentation is required wherever the CI will be operated, maintained, repaired, trained, supported, and reprocured. A significant factor in this determination is data that is properly established as "Contractor Proprietary." Authority rests with the program manager to decide whether it is necessary and cost effective to buy rights to the data, do without it, develop new data CIs, or return to the original contractor whenever reprocurement or support of the CI is needed.

Engineering Release: Program managers should ensure that both contractors and DoD activities follow engineering release procedures, which record the release and retain records of approved configuration documentation. These records ensure:

- An audit trail of CI documentation status and history
- Verification that engineering documentation has been changed to reflect the incorporation of approved changes and to satisfy the requirements for traceability of deviations and engineering changes
- A means to reconcile engineering and manufacturing data to assure that engineering changes have been accomplished and incorporated into deliverable units of the CIs

Interface Management: Program managers normally have interface requirements with other systems. Those interfaces constitute design constraints imposed on the programs. As the system is defined, other interfaces between system components become apparent. All interfaces need to be identified and documented, e.g., Interface Control Documents (ICDs), so that their integrity may be maintained through a disciplined configuration control process.

Supportability: AR initiatives, such as the emphasis on COTS/NDIs, have changed the traditional methods of supporting a system during the production and operational phases. The system supportability concept is a major decision factor in determining the extent of the DoD's CM involvement and determining the extent of the TDP CDRL requirement. The following should be considered when planning for system supportability, maintenance, and risk reduction:

- The RFP, proposal preparation instructions (section L) should have CM as a key management and past performance discriminator. The weighting of the RFP evaluation criteria (section M) should reflect the importance of an effective, documented contractor CM process as a risk mitigator
- Interface, interoperability, and coordination requirements are defined for the LRUs/parts consistent with the maintenance philosophy
- The maintenance plan is a primary driver for the level of configuration control and support requirements. Coordinating CM requirements with the maintenance plan (support and maintenance planning) and logistics personnel is imperative
- Program manager retains configuration change control authority on changes that impact compatability; life; reliability; interchangeability, form, fit, function (F^3) and safety
- Program plans and budgets should include early planning for purchase of the TDP as appropriate
- Items provided under a performance specification at different times or from different suppliers should be interchangeable, but may not be identical in internal design. Where appropriate, bidders should be provided with the specific dimensional, material, manufacturing and assembly information needed to supply identical items with each reprocurement
- When the commercial items ordered or offered have been wholly or partially developed with private funding, the commercial supplier is generally willing to only provide F^3 information. This information includes such items as brochures, operating and training manuals, and organizational maintenance technical manuals. Suppliers are generally not willing to provide the Government with the design and manufacturing data necessary for a competitor to build the same product in quantity or to conduct major repairs or rebuilds
- Technical Data Packages contain detail configuration data down to the lowest replaceable/repairable units (LRU) or parts consistent with the maintenance philosophy
- Consider purchasing the Technical Data Package when the following apply:
  - If upgrades and follow-ons to the system will be open bid, with the possibility of another contractor being a prime for follow-on contracts

- Dual depots are used for maintenance
- The system is a largely COTS/NDI system, which will normally require technology refresh

APPENDIX A

Extracts of Risk Management Requirements in the DoD 5000 Series Documents

(Issued 15 March 1996)

DoDD 5000.1 Defense Acquisition

a. Section D, paragraph 1.d. Risk Assessment and Management
PMs and other acquisition managers shall continually assess program risks. Risks must be well understood, and risk management approaches developed, before decision authorities can authorize a program to proceed into the next phase of the acquisition process. To assess and manage risk, PMs and other acquisition managers shall use a variety of techniques, including technology demonstrations, prototyping, and test and evaluation. Risk management encompasses identification, mitigation, and continuous tracking, and control procedures that feed back through the program assessment process to decision authorities. To ensure an equitable and sensible allocation of risk between Government and industry, PMs and other acquisition managers shall develop a contracting approach appropriate to the type of system being acquired.

DoD 5000.2-R Mandatory Procedures for Major Defense Acquisition Programs (MDAPs) and Major Automated Information System (MAIS) Acquisition Programs

a. Part 1, Section 1.1 Purpose
This part establishes a general model for ... acquisition programs. The ... model acknowledges that every acquisition program is different. PM and MDA shall structure the (program) to ensure a logical progression through a series of phases designed to reduce risk, ... and provide adequate information for decision-making ....

b. Part 1, Section 1.2 Overview of the Acquisition Management Process
The acquisition process shall be structured in logical phases separated by major decision points called milestones .... Threat projections, system performance ... and risk management shall be major considerations at each milestone decision point, including the decision to start a new program.

c. Part 1, Section 1.4.2 Phase 0: Concept Exploration
Phase 0 typically consists of ... short-term concept studies. The focus ...is to define and evaluate the feasibility of alternative concepts and to provide a basis for assessing the relative merits (i.e., advantages and disadvantages, degree of risk) of these concepts at the next milestone decision point...

d. Part 1, Section 1.4.3 Phase 1: Program Definition and Risk Reduction
During this phase, the program shall become defined as one or more concepts, design approaches, and/or parallel technologies are pursued as warranted. Assessments ... shall be refined. Prototyping, demonstrations, and early operational assessments shall be considered and included as necessary to reduce

risk so that technology, manufacturing, and support risks are well in hand before the next decision point...

e. Part 2, Section 2.3 Requirements Evolution
Thresholds and objectives are defined below. The values for an objective or threshold and definitions for any specific parameter contained in the ORD, TEMP, and APB shall be consistent.

1. Threshold. ...is minimum acceptable value ... to satisfy the need .... The spread between objective and threshold values shall be individually set for each program based on characteristics of the program (e.g., maturity, risk, etc.).
2. Objective. ... that (value) desired by the user and which the PM is attempting to obtain...

f. Part 3, Section 3.2.2.2 APB Content
The APB shall contain only the most important cost, schedule and performance parameters ...

1. Performance...
2. Schedule...
3. Cost. In all cases, the cost parameters shall reflect the total program and be realistic cost estimates, based on a careful assessment of risks and realistic appraisals of the level of costs most likely to be realized...

g. Part 3, Section 3.2.3 Exit Criteria
MDAs shall use exit criteria to establish goals .... Exit criteria will normally be selected to track progress in important technical, schedule, or management risk areas.

h. Part 3, Section 3.3 Acquisition Strategy
Each PM shall develop and document an acquisition strategy ... roadmap for program execution... Essential elements include ... risk management ....

i. Part 3, Section 3.3.1.3 Industrial Capability
The PM shall structure the acquisition strategy to promote sufficient program stability to encourage industry to invest, plan and bear risks...

The program acquisition strategy shall analyze the industrial capability to design, develop, produce, support.... This analysis shall identify DoD investments needed to create new industrial capabilities and the risks of industry being unable to provide program manufacturing capabilities at planned cost and schedule...

j. Part 3, Section 3.3.2 Cost, Schedule, and Performance Risk Management
The PM shall establish a risk management program for each acquisition program to identify and control performance, cost and schedule risks. The risk management program shall identify and track risk drivers, define risk abatement plans, and provide for continuous risk assessment throughout each acquisition phase to determine how risks have changed. Risk reduction measures shall be

included in cost-performance trade-offs, where applicable. The risk management program shall plan for back-ups in risk areas and identify design requirements where performance increase is small relative to cost, schedule, and performance risk. The acquisition strategy shall include identification of the risk areas of the program and a discussion of how the PM intends to manage those risks.

k. Part 3, Section 3.3.3.2 Cost Management Incentives
RFPs shall be structured to incentivize the contractor to meet or exceed cost objectives. Whenever applicable, risk reduction through use of mature processes shall be a significant factor in source selection...

l. Part 3, Section 3.3.4 Contract Approach
The acquisition strategy shall discuss the types of contracts contemplated for each succeeding phase, including considerations of risk assessment, reasonable risk-sharing by Government and contractor(s)...

m. Part 3, Section 3.3.4.1 Competition
PMs and contracting officers shall provide for full and open competition, unless one of the limited statutory exceptions apply...

The PM shall consider component breakout. (which) shall be done when there are significant cost savings, ... when the technical or schedule risk of furnishing Government items to the prime contractor is manageable...

n. Part 3, Section 3.3.5.6 Information Sharing and DoD Oversight
DoD oversight activities ... shall consider all relevant and credible information that might mitigate risks and the need for DoD oversight...

o. Part 3, Section 3.4 Test and Evaluation
Test and evaluation programs shall be structured to integrate all (test and evaluation) activities conducted by different agencies as an efficient continuum. All such activities shall be part of a strategy to provide information regarding risk and risk mitigation...

p. Part 3, Section 3.4.1 Test and Evaluation Strategy
Test and evaluation planning shall begin in Phase 0, Concept Exploration...

6. Early testing of prototypes in Phase I, Program Definition and Risk Reduction, and early operational assessments shall be emphasized to assist in identifying risks.

q. Part 3, Section 3.4.2 Development Test and Evaluation
Development test and evaluation (DT&E) programs shall...

3. Support the identification and description of design technical risks;
4. Assess progress toward meeting Critical Operational Issues, mitigation of acquisition technical risk, achievement of manufacturing process requirements and system maturity...

r. Part 3, Section 3.4.3 Certification of Readiness for Operational Test and Evaluation

In support of this [certification], risk management measures and indicators, with associated thresholds, which address performance and technical adequacy of both hardware and software shall be defined and used on each program. A mission impact analysis of criteria and threshold that have not been met shall be completed prior to certification for operational tests.

s. Part 3, Section 3.5.1 Life Cycle Cost Estimates

The life cycle cost estimates shall be:

4. Neither optimistic nor pessimistic, but based on a careful assessment of risks and reflecting a realistic appraisal of the level of cost most likely to be realized.

t. Part 4, Section 4.2 Integrated Process and Product Development

It is critical that the processes used to manage, develop, manufacture, verify, test, deploy, operate, support, train people, and eventually dispose of the system be considered during program design.

u. Part 4, Section 4.3 Systems Engineering

The PM shall ensure that a systems engineering process is used to translate operational needs and/or requirements into a system solution that includes the design, manufacturing, test and evaluation, and support processes and products. The systems engineering process shall establish a proper balance between performance, risk, cost, and schedule...

The systems engineering process shall: ...

3. Characterize and manage technical risks.

The key systems engineering activities that shall be performed are:

4. System Analysis and Control. System analysis and control activities shall be established to serve as a basis for evaluating and selecting alternatives, measuring progress, and documenting design decisions. This shall include:
    b. The establishment of a risk management process to be applied throughout the design process. The risk management effort shall address the identification and evaluation of potential sources of technical risks based on the technology being used and its related design, manufacturing, test and support processes, risk mitigation efforts, and risk assessment and analysis. Technology transition planning and criteria shall be established as part of the overall risk management effort.

The following areas reflect important consideration in the design and shall be a part of the systems engineering process. The extent of their consideration and impact on the product design shall be based on the degree to which they impact total system cost, schedule and performance, at an acceptable level of risk.

4.3.1 Manufacturing and Production

4.3.2 Quality

4.3.3 Acquisition Logistics

4.3.4 Open Systems Design

4.3.5 Software Engineering

4.3.6 Reliability, Maintainability and Availability

4.3.7 Environment, Safety, and Health

4.3.8 Human Systems Integration (HSI)

4.3.9 Interoperability

SECNAV Instruction 5000.2B implements the requirements of the DoD 5000 Series.

✍ ***PM Note***

APPENDIX B

Additional Sources of Information

Introduction

Chapter 5 provided critical design, test and production processes as an aid in performing risk assessments. These processes are only intended to be used as a starting point from which programs can expand with their own critical processes, tailored to their unique program needs. As an additional aid, this Appendix provides sources of information sponsored by DoD to assist in the dissemination of scientific and technical information, i.e., the Information Analysis Centers (IACs) chartered by DoD and the manufacturing Centers of Excellence (COEs) sponsored by the ManTech Programs of the Army, Navy, Air Force, and Defense Logistics Agency (DLA).

DoD Information Analysis Centers

DoD IACs are formal organizations chartered by DoD to facilitate utilization of existing scientific and technical information.

The primary mission of DoD IACs is to collect, analyze, synthesize, and disseminate worldwide scientific and technical information in clearly defined, specialized fields or subject areas. A secondary mission is to promote standardization within their respective fields. The IACs have a broad mission to improve the productivity of scientists, engineers, managers, and technicians in the Defense community through timely dissemination of evaluated information.

Mission and functions of the IACs

Thirteen contractor-operated DoD IACs are administratively managed and funded by the Defense Technical Information Center (DTIC). Eleven other IACs are managed by the Services. Individual IACs may be contacted directly for information requiring technical expertise or expert judgment in their particular area. A listing of each IAC and on-line address information is provided below. However, most of the DoD and Service sponsored IACs may be contacted by sending an e-mail message to:

dodiacs@dtic.mil

Access most IACs on-line at the following address

DTIC IACs:

- Advanced Materials and Processes Technology Information Analysis Center (AMPTIAC) **http://rome.iitri.com/amptiac**

- Chemical Warfare/Chemical & Biological Defense IAC (CBIAC) **http://www.cbiac.apgea.army.mil**

  Chemical Propulsion Information Agency (CPIA) **http://www.jhu.edu/~cpia**

- Crew System Ergonomics Information Analysis Center (CSERIAC) **http://cseriac.flight.wpafb.af.mil**

- Data and Analysis Center for Software (DACS) **http://www.dacs.com**
- Defense Modeling, Simulation, and Tactical Technology Information Analysis Center (DMSTTIAC) **http://dmsttiac.hq.iitri.com**
- Guidance and Control Information Analysis Center (GACIAC) **http://gaciac.hq.iitri.com**
- Information Assurance Technology Analysis Center (IATAC) **http://www.iatac.dtic.mil**
- Infrared Information Analysis Center (IRIAC) **http://www.erim.org/IRIA/iria.html**
- Manufacturing Technology Information Analysis Center (MTIAC) **http://www.mtiac.iitri.com**
- Nondestructive Testing Information Analysis Center (NTIAC) **http://www.ntiac.com**
- Reliability Analysis Center (RAC) **http://rome.iitri.com/rac**
- Survivability/Vulnerability Information Analysis Center (SURVIAC) **http://surviac.flight.wpafb.af.mil**

Service IACs:

- Aerospace Structures Information Analysis Center (ASIAC) E-mail: siac@fltvc1.flight.wpafb.af.mil
- Supportability Investment Decision Analysis Center (SIDAC) **http://www.sidac.wpafb.af.mil**
- Airfields, Pavements, and Mobility Information Analysis Center (APMIAC) E-mail: wesgva@ex1.wes.army.mil
- Coastal Engineering Defense Information Analysis Center (CEIAC) E-mail: s.wagner@cerc.wes.army.mil
- Cold Regions Science and Technology Information Analysis Center (CRSTIAC) **http://www.crrel.usace.army.mil/crstiac**
- Concrete Technology Information Analysis Center (CTIAC) E-mail: matherb@ex1.wes.army.mil

- Environmental Information Analysis Center (EIAC)
  http://www.wes.army.mil/el/homepage.html
- Hydraulic Engineering Information Analysis Center (HEIAC)
  http://hlnet.wes.army.mil
- Soil Mechanics Information Analysis Center (SMIAC)
  http://www.wes.army.mil/GL/SMIAC/smiac.html
- Shock and Vibration Information Analysis Center (SAVIAC)
  http://saviac.usae.bah.com
- DoD Nuclear Information Center (DASIAC)

Manufacturing Centers Of Excellence

Mission and functions of the COEs

The manufacturing Centers Of Excellence (COE's) sponsored by the ManTech Programs of the Army, Navy, Air Force and DLA provide a focal point for the development and transfer of new manufacturing processes and equipment in a cooperative environment with industry, academia and DoD activities. The COEs:

- Develop and demonstrate manufacturing technology solutions for identified defense manufacturing issues
- Serve as corporate residences of expertise in their particular technological areas
- Provide consulting services to defense industrial activities and industry
- Facilitate the transfer of developed manufacturing technology; and
- Provide advice to the ManTech Program directors concerning program formulation

One stop address for all COEs

The COEs have been set up in consortium-type arrangements wherein industry, academia, and Government can be involved in developing and implementing advanced manufacturing technologies. An overview of each COE is available at:

http://mantech.iitri.com/program/centexel.html

The following is a list of the centers:

- Apparel Manufacturing Demonstration Center
- Best Manufacturing Practices Center of Excellence (BMPCOE)
- Center for Optics Manufacturing (COM)
- Center of Excellence for Composites Manufacturing Technology (CECMT)
- Combat Rations Demonstration Center
- Electronics Manufacturing Productivity Facility (EMPF)
- Energetics Manufacturing Technology Center (EMTC)

- Gulf Coast Region Maritime Technology Center (GCRMTC)
- Instrumented Factory for Gears (INFAC)
- Manufacturing Science & Advanced Materials Processing Institute (MS&AMPI), Composed of three centers:
  - Laser Applications Research Center (LaserARC)
  - National Center for Advanced Drivetrain Technologies (NCADT)
  - Surface Engineering Manufacturing Technology Center (SEMTC)
- Manufacturing Technology Transfer Center (MTTC)
- National Center for Excellence in Metalworking Technology (NCEMT)
- National Center for Manufacturing Sciences (NCMS)
- National Network for Electro-Optics Manufacturing Technology (NNEOMT)
- Navy Joining Center (NJC)

Ordering Information

Additional copies of NAVSO P-3686 are available on CD-ROM.
Submit requests for additional copies to:

Office of the Assistant Secretary of the Navy
(Research, Development and Acquisition)
Acquisition and Business Management
Crystal Plaza 5, Room 568
2211 South Clark Place
Arlington, VA 22244-5104

Orders may also be placed via Fax to:
Sandy Kraft at 703-602-3129

or copies may be downloaded from the ASN (RD&A) ABM Home Page at:
http://www.abm.rda.hq.navy.mil