SYSTEMS ENGINEERING LEADING INDICATORS GUIDE

Version 2.0

January 29, 2010

Supersedes Initial Release, June 2007

Editors

Garry Roedler
Lockheed Martin Corporation
garry.j.roedler@lmco.com

Donna H. Rhodes
Massachusetts Institute of Technology
rhodes@mit.edu

Howard Schimmoller
Lockheed Martin Corporation
howard.j.schimmoller@lmco.com

Cheryl Jones
US Army
cheryl.jones5@us.army.mil

Developed and Published by Members of

INCOSE Technical Product Number: INCOSE-TP-2005-001-03

Additional collaboration for this version:

For Version 2.0, the following major organizations, agencies, and initiatives were contributors to the revision:

- BAE Systems
- Boeing
- Defense Contract Management Agency
- International Council on Systems Engineering
- General Dynamics
- Lockheed Martin
- Massachusetts Institute of Technology
- Massachusetts Institute of Technology – Lean Advancement Initiative
- Massachusetts Institute of Technology – Systems Engineering Advancement Research Initiative
- MITRE
- National Defense Industrial Association (NDIA) Systems Engineering Division (SED)
- Naval Air System Command
- Northrop Grumman Corporation (NGC)
- Practical Software and Systems Measurement
- PRICE Systems
- Raytheon
- Science Applications International Corporation (SAIC)
- Systems Engineering Research Center
- Third Millennium Systems
- University of Southern California
- US Air Force Center for Systems Engineering
- US Army Research, Development, Engineering Command-Armament Research Development & Engineering Center (RDECOM-ARDEC)
- US Office of Secretary of Defense

Table of Contents

1 ABOUT THIS DOCUMENT

This Systems Engineering Leading Indicators Guide is the result of a project initiated by the MIT Lean Advancement Initiative (LAI) in cooperation with the International Council on Systems Engineering (INCOSE), Practical Software and Systems Measurement (PSM), MIT Systems Engineering Advancement Research Initiative (SEAri), Naval Air Systems Command (NAVAIR), and the Department of Defense Systems Engineering Research Center (SERC). Leading measurement and systems engineering experts from government, industry, and academia volunteered their time to work on this initiative. The Systems Engineering Leading Indicators Guide is issued by INCOSE as document number INCOSE-TP-2005-001-03. This document is also available from the PSM website and MIT LAI and MIT SEAri websites.

Government and industry organizations are encouraged to tailor the information in this document for their purposes, and may incorporate this material into internal guidance documents. Please cite the original source and release version for traceability and baseline control purposes.

Contacts:

For further information, contact the SE Leading Indicators project leaders: Donna Rhodes – Massachusetts Institute of Technology (rhodes@mit.edu), Garry Roedler - Lockheed Martin (garry.j.roedler@lmco.com), Cheryl Jones - US Army/RDECOM-ARDEC (cheryl.jones5@us.army.mil), or Howard Schimmoller – Lockheed Martin (howard.j.schimmoller@lmco.com). Gregory Hein - NAVAIR (AIR-4.1)/Booz Allen Hamilton (gregory.hein.ctr@navy.mil) is the primary point of contact for Appendix A – NAVAIR Applied Leading Indicator.

Contributors:

LEADERSHIP TEAM

Cheryl Jones	US Army RDECOM-ARDEC/Practical Systems & Software Measurement (PSM)
Garry Roedler	Lockheed Martin/INCOSE/PSM/NDIA SED
Donna Rhodes	Massachusetts Institute of Technology/LAI/SEAri/INCOSE
Howard Schimmoller	Lockheed Martin

CORE TEAM

Dennis Ahern	Northrop Grumman	Robert Koury	PRICE Systems
Peter Baxter	Distributive Management	Dave Henry	Lockheed Martin/INCOSE
Winsor Brown	University of Southern California/SERC	Trindy LeForge	Raytheon
Barry Boehm	University of Southern California/SERC	Chris Miller	US Office of Secretary of Defense/SAIC/INCOSE
Reggie Cole	Lockheed Martin	John Miller	Lockheed Martin
Richard Cline	Boeing	Gregory Niemann	Lockheed Martin
Tina Denq	NAVAIR (AIR-4.1)/Booz Allen Hamilton	Juan Ortiz	NAVAIR (AIR-4.1) SEDIC Director
Susan Ferreira	University of Texas at Arlington	John Rieff	Raytheon/INCOSE
Paul Frenz	General Dynamics/INCOSE	Carlos Shaeffers	DCMA
John Gaffney	Lockheed Martin	Sarah Sheard	Third Millennium Systems/INCOSE
Kishore Gagrani	PRICE Systems	Robert Swarz	MITRE
Don Gantzer	SAIC(SESO)/INCOSE MWG	Mike Ucchino	Air Force Center for Systems Engineering/INCOSE/NDIA SED
William Golaz	Lockheed Martin	Ricardo Valerdi	Massachusetts Institute of Technology/LAI/SEAri/INCOSE
Gregory Hein	NAVAIR (AIR-4.1)/Booz Allen Hamilton	Gan Wang	BAE Systems/INCOSE
Dan Ingold	University of Southern California/SERC	Bob Welldon	BAE Systems

PREVIOUS CONTRIBUTORS

Daniel Abbott	Air Force	Jeff Loren	SAF/AQRE
David Card	Q-Labs	William Miller	Stevens Institute of Technology
Ron Carson	Boeing/INCOSE	Mark Mithers	Northrop Grumman
Bruce Evans	Lockheed Martin	Rick Neupert	Boeing
Kacy Gerst	Massachusetts Institute of Technology/SEAri	Paul Robitaille	Lockheed Martin/INCOSE
Sheree Havlik	Raytheon	Darlene Utter	Massachusetts Institute of Technology/LAI Student
Tom Huynh	Naval Postgraduate School	Mark Wilson	Air Force Center for Systems Engineering
Caroline Lamb	Massachusetts Institute of Technology/LAI Student	Michael Winter	Pratt & Whitney

Additional thanks goes to many additional colleagues who contributed to the development of the guide thorough participation in meetings, workshops and reviews.

2 EXECUTIVE SUMMARY

Several policies calling for improved systems engineering on programs were released by DoD and the services during 2004[1]. During this period, the Lean Advancement Initiative (LAI) Consortium (now the Lean Advancement Initiative) was tasked with assisting with the systems engineering revitalization activity. In June 2004, an *Air Force/LAI Workshop on Systems Engineering for Robustness*[2] was held to establish the groundwork for several initiatives in support of systems engineering revitalization. One of these initiatives focused on ***leading indicators for evaluating the goodness of systems engineering on a program***. This initiative was jointly supported by LAI, INCOSE, PSM, and others in an industry collaborative effort to address this need. In December 2005, the beta version of this document was released, describing the initial set of SE Leading Indicators. This initial set reflected the subset of possible candidate indicators that were considered to be the highest priority by the team, recognizing that the set was not exhaustive. In June 2007, Version 1.0 of this document was released following the completion of a validation phase which included pilot applications of the leading indicators, a research study, various workshops, and an industry survey. In this 2010 revision, additional indicators were added, and further implementation recommendations and guidance on interpretation has been provided. Additional SE Leading Indicators will be added in future updates as these are identified, defined, and evolved.

What are SE Leading Indicators? A leading indicator is a measure for evaluating the effectiveness of a how a specific activity is applied on a project in a manner that provides information about impacts that are likely to affect the system performance objectives. A leading indicator may be an individual measure, or collection of measures and associated analysis that are predictive of future systems engineering performance before the system is fully realized. Systems engineering performance itself could be an indicator of future project execution and system performance. Leading indicators aid leadership in delivering value to customers and end users, while assisting in taking interventions and actions to avoid rework and wasted effort.

Who Developed the SE Leading Indicators? Subsequent to the June 2004 workshop, the "SE Leading Indicators Action Team" was formed under the auspices of LAI, comprised of engineering measurement experts from industry, government and academia, involving a collaborative partnership with INCOSE[3] . Mr. Garry Roedler of Lockheed Martin and Dr. Donna Rhodes of MIT co-led the effort. Leading SE and measurement experts from LAI member companies, INCOSE, SSCI[4], and PSM[5] volunteered to serve on the team. The team held periodic meetings and used the ISO/IEC 15939 and PSM Information Model to define the indicators. The original version had thirteen SE leading indicators. Based on feedback from users, surveys, and workshops, a decision was made to revise the guide to add indicators and application information, as well as improve usability. For this revision, additional collaborative partners joined the team, including, NAVAIR, NDIA SED, and DoD SERC. Both versions have

---

[1] Policies include Policy for Systems Engineering in the DOD, 20 Feb 04; Assistant Secretary of the Air Force for Acquisition, Dr Sambur , 9 Apr 03, Policy Memo 03A-005 titled Incentivizing Contractors for Better Systems Engineering; Memo 04A-001 titled Revitalizing AF and Industry Systems Engineering Increment 2

[2] Rhodes, D. Ed, Report on the AF/LAI Workshop on Systems Engineering for Robustness, July 2004, http://lean.mit.edu

[3] INCOSE (International Council on Systems Engineering) is the leading professional society for systems engineering. INCOSE has developed guidance materials on systems engineering measures, and both editors of document have served as former chairs of the INCOSE Measurement Working Group. INCOSE is collaborating with LAI on this effort, and is targeted as the long term owner for guidance developed under this LAI project.

[4] SSCI (Systems and Software Consortium Inc.) is collaborating with LAI on systems engineering initiatives.

[5] PSM (Practice Software and Systems Measurement) has developed foundational work on measurements under government funding. The LAI effort is using formats developed by PSM for documenting of the leading indicators.

had significant support from a large number of industry partners identified in the contributors section of this guide. To date, eighteen SE leading indicators have been developed, as summarized in Table 1.

What Problem do SE Leading Indicators Address? Leading indicators support the effective management of systems engineering by providing visibility into expected project performance and potential future states. Visibility into the future state of a project has not traditionally been part of a measurement process. Additionally, without the use of leading indicators, it is difficult for leadership to establish the likelihood of delivering a complex system within the project constraints such as scope, schedule, quality and budget.

Who are the Primary Users of the SE Leading Indicators? The primary users are the project specific systems engineering leadership, project management, and IPT leadership who use the indicators to assess and make adjustments for assuring systems engineering effectiveness of the project. Selected indicators may also be used by the project customers, project partners, and project suppliers depending on phase of project and nature of the contractual relationship. Secondary users include executive and business area management, as well as process owners, for the purpose of predicting the overall effectiveness of systems engineering within and across a project, and for early detection of problems that require management attention.

How do SE Leading Indicators Differ from Conventional SE Measures? A conventional measure provides insight into the issue areas of interest to management using historic and current status information. In contrast a leading indicator draws on trend information of conventional measures or significant correlation to provide predicative analysis. A leading indicator, for example, could use requirements growth to predict the future behavior of another process or sub-process. While the data on which both conventional measurement and leading indicators is similar, a key difference is that leading indicators address information needs that are predictive or forward looking. While the leading indicators appear similar to existing measures and often use the same base information, ***the difference lies in how the information is gathered, evaluated, interpreted, and used to provide a forward looking perspective*****.**

How do SE Leading Indicators relate to Current Organizational SE Measurement Practices? Most organizations have an organizational measurement plan and a set of measures. These leading indicators are meant to augment the existing set of measures. For optimal efficiency these should be implemented via the organization's measurement infrastructure (typically based on CMMI® practices), thereby enabling mechanized data gathering, analysis, and evaluation. It should also be noted that leading indicators involve use of empirical data to set planned targets and thresholds. Where organizations lack this data, expert opinion may be used as a proxy to establish initial targets and thresholds until a good historical base of information can be collected, but should not be relied on as a long term solution for measurement projections. Rather, organizations must build the collection of the historical measurement data into its collection practices.

What is the Expected Impact? These leading indicators have been specifically selected to provide insight into key systems engineering activities across the phases of a project.

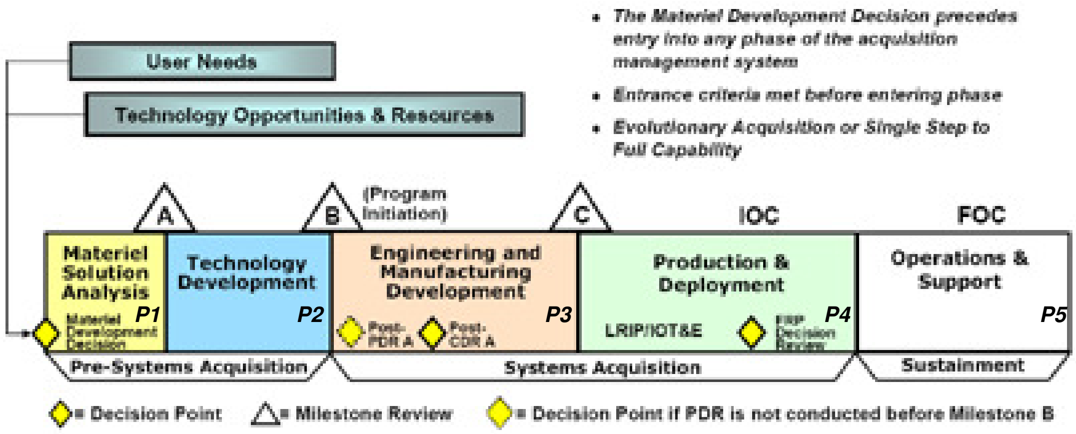


Figure 1 - The Defense Acquisition Management Framework

Figure 1 - The Defense Acquisition Management Framework[6], depicts the United States Department of Defense (DoD) acquisition life cycle phases for a defense program. These phases were established and described by DoD Instruction 5000.02 and the associated Defense Acquisition Guidebook. This process is a continuum of activities for managing all defense acquisition programs. Appropriate tailoring of the detailed measurement information specifications may be needed to address the specific information needs of any given program. It should be noted that the leading indicators are also envisioned as suitable to commercial endeavors.

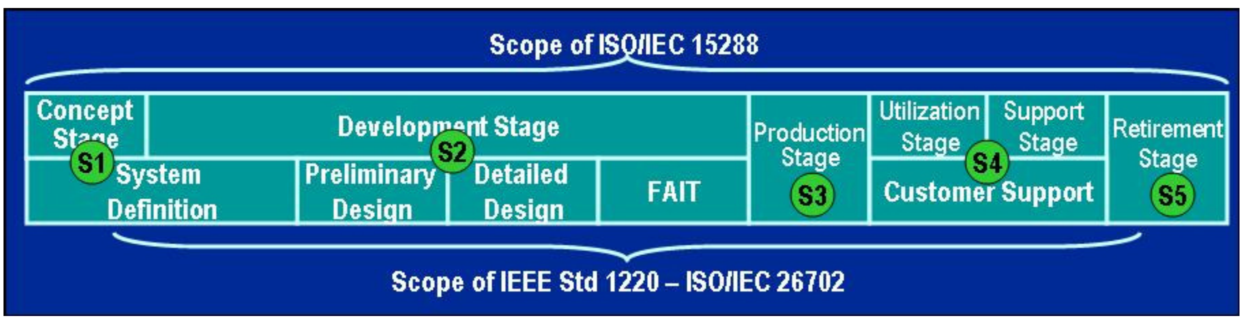


Figure 2 - ISO/IEC 15288, IEEE 1220 and ISO/IEC 26702 Stages

Figure 2 - ISO/IEC 15288, IEEE 1220 and ISO/IEC 26702 Stages, depict the same process and life cycle concepts in non-DoD terms. As demonstrated in Table 1, most of the leading indicators are trend measures that have broad applicability across both defense and commercial life cycle phases/stages. The DoD life cycle phases are denoted as P1 through P5 in Table 1 and are numbered as P1-P5 in Figure 1. The industry standard life cycle stages are denoted as S1 through S5 in both Table 1 and Figure 2. Table 2 further refines the applicability of each leading indicator to the associated ISO/IEC 15288 activities.

What is an example of how SE Leading Indicators have contributed to effective systems engineering on a project? A good example of the positive impact of using leading indicators was demonstrated within one of the pilots of the beta release guide. By monitoring the requirements validation and volatility trends, the pilot project team was able to more effectively predict readiness for the System Requirements Review (SRR) milestone. Initially the project had selected a calendar date to conduct the SRR, but in subsequent planning made the decision to have the SRR be event driven,

[6] Interim Defense Acquisition Guide, Dec 2009 Draft, Nov 14 2009, http://akss.dau.mil/dag

resulting in a new date for the review wherein there could be a successful review outcome. That is, the review date was set based on an acceptable level of requirements validation and volatility in accordance with the leading indicators. Had the original calendar date been used, it is likely that the SRR would not have been successful and would have had to be repeated. See the example "Requirements Volatility" graphic.

Are the SE Leading Indicators Applicable to System of Systems Programs? The leading indicators have primarily been derived from experience on traditional projects, however potential for use on System of Systems (SoS) projects has been given some consideration. First of all, some of the leading indicators are directly usable by a prime contractor as indicators for SoS level engineering activities. As SoS projects apply many of the same skills and perform many of the same activities as systems projects, the leading indicators do still apply. It is anticipated that in the SoS case, the interpretation of the leading indicators may involve some additional and/or unique considerations. For example how leading indicators, applied at the constituent systems level of a SoS, could be used effectively as a collected set of indicators and/or as aggregated indicators.

How will the SE Leading Indicators be Further Validated? The further validation efforts will be monitored by the core team, in collaboration with the participating collaboration organizations. Based on results of project use, leading indicators will be adjusted as required. Additionally, recommendations will be developed regarding which leading indicators are most effective for particular types of projects.

What are the Plans for Improvement? In support of the continuing validation and refinement activity, industry and academic research is ongoing and planned to analyze the effectiveness and adequacy of the measures in support of improved project performance. As lessons are learned in the continuing validation process, the core team will be providing briefings to and seeking input from selected government forums and systems engineering societies/associations. There are several activities planned for the future, including workshops on leading indicators involving cross discipline participation.

The ongoing maintenance of this guidance will be facilitated collaboratively by a leadership team from INCOSE, MIT, and PSM. This leadership team will meet on a regular basis, leveraging existing functions such as the INCOSE International Workshop, PSM Users Group Conference, and LAI Annual Conference in order to involve as many of the collaborative partners as possible. The ongoing work will include workshops with the contributors and interested parties to examine results of the indicators and opportunities for improvement. Consideration will be given to establishing a website to facilitate ongoing communication of contributors and users.

Table 1 - SYSTEMS ENGINEERING LEADING INDICATORS OVERVIEW											
Leading Indicator	**Insight Provided**	**Phases / Stages**									
		P 1	**P 2**	**P 3**	**P 4**	**P 5**	**S 1**	**S 2**	**S 3**	**S 4**	**S 5**
Requirements Trends	Rate of maturity of the system definition against the plan. Additionally, characterizes the stability and completeness of the system requirements that could potentially impact design, production, operational utility, or support.	●	●	●	●	●	●	●	●	●	●
System Definition Change Backlog Trend	Change request backlog which, when excessive, could have adverse impact on the technical, cost and schedule baselines.			●	●	●		●	●	●	
Interface Trends	Interface specification closure against plan. Lack of timely closure could pose adverse impact to system architecture, design, implementation and/or V&V any of which could pose technical, cost and schedule impact.	●	●	●	●	●	●	●	●	●	
Requirements Validation Trends	Progress against plan in assuring that the customer requirements are valid and properly understood. Adverse trends would pose impacts to system design activity with corresponding impacts to technical, cost & schedule baselines and customer satisfaction.	●	●	●	●	●	●	●	●	●	
Requirements Verification Trends	Progress against plan in verifying that the design meets the specified requirements. Adverse trends would indicate inadequate design and rework that could impact technical, cost and schedule baselines. Also, potential adverse operational effectiveness of the system.	●	●	●	●	●	●	●	●	●	●
Work Product Approval Trends	Adequacy of internal processes for the work being performed and also the adequacy of the document review process, both internal and external to the organization. High reject count would suggest poor quality work or a poor document review process each of which could have adverse cost, schedule and customer satisfaction impact.	●	●	●	●	●	●	●	●	●	
Review Action Closure Trends	Responsiveness of the organization in closing post-review actions. Adverse trends could forecast potential technical, cost and schedule baseline issues.	●	●	●	●	●	●	●	●	●	●
Risk Exposure Trends	Effectiveness of risk management process in managing / mitigating technical, cost & schedule risks. An effective risk handing process will lower risk exposure trends.	●	●	●	●	●	●	●	●	●	●
Risk Treatment Trends	Effectiveness of the SE organization in implementing risk mitigation activities. If the SE organization is not retiring risk in a timely manner, additional resources can be allocated before additional problems are created.	●	●	●	●	●	●	●	●	●	●
Technology Maturity Trends	Risk associated with incorporation of new technology or failure to refresh dated technology. Adoption of immature technology could introduce significant risk during development while failure to refresh dates technology could have operational effectiveness/customer satisfaction impact.		●	●	●	●		●	●	●	
Technical Measurement Trends	Progress towards meeting the Measures of Effectiveness (MOEs) / Performance (MOPs) / Key Performance Parameters (KPPs) and Technical Performance Measures (TPMs). Lack of timely closure is an indicator of performance deficiencies in the product design and/or project team's performance.			●				●			
Systems Engineering Staffing & Skills Trends	Quantity and quality of SE personnel assigned, the skill and seniority mix, and the time phasing of their application throughout the project lifecycle.	●	●	●	●	●	●	●	●	●	●

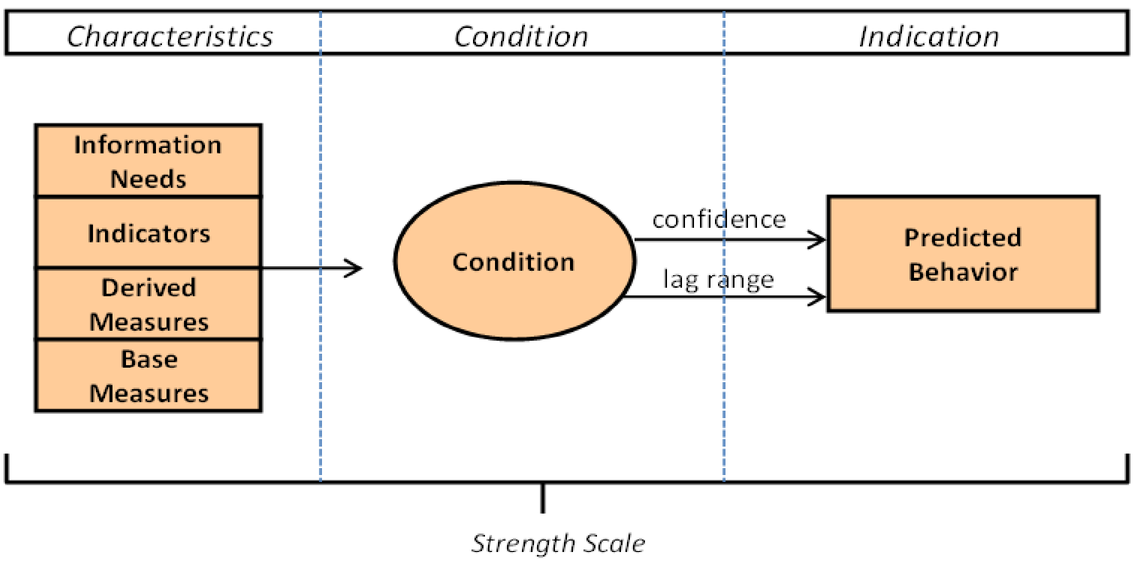
Table 1 - SYSTEMS ENGINEERING LEADING INDICATORS OVERVIEW											
Leading Indicator	**Insight Provided**	**Phases / Stages**									
		P 1	**P 2**	**P 3**	**P 4**	**P 5**	**S 1**	**S 2**	**S 3**	**S 4**	**S 5**
Process Compliance Trends	Quality and consistency of the project defined SE process as documented in SEP/SEMP. Poor/inconsistent SE processes and/or failure to adhere to SEP/SEMP, increase project risk.	●	●	●	●	●	●	●	●	●	●
Facility and Equipment Availability Trends	Availability of non-personnel resources (infrastructure, capital assets, etc.) needed throughout the project lifecycle.	●	●	●	●	●	●	●	●	●	●
Defect/Error Trends	Progress towards the creation of a product or the delivery of a service that meets the quality expectations of its recipient. Understanding the proportion of defects being found and opportunities for finding defects at each stage of the development process of a product or the execution of a service.	●	●	●	●	●	●	●	●	●	●
System Affordability Trends	Progress towards a system that is affordable for the stakeholders. Understanding the balance between performance, cost, and schedule and the associated confidence or risk.	●	●	●	●	●	●	●	●	●	●
Architecture Trends	Maturity of an organization with regards to implementation and deployment of an architecture process that is based on an accept set of industry standards and guidelines.	●	●	●			●	●			
Schedule and Cost Pressure	Impact of schedule and cost challenges on carrying out a project	●	●	●	●	●	●	●	●	●	●

3 LEADING INDICATORS

The following subsections provide a description of the leading indicators. Each description provides a rationale that justifies the value of the indicator, describes the decision insight it offers, and specifies how to measure and calculate it. In addition, there are sample graphics intended to illustrate the use of the indicator.

Important Note: *The graphics in this document are intended for basic illustrative purpose only, and may represent only one aspect of the overall indicator. These are notional graphs and do not contain actual data. A conscience effort has been made to depict the leading indicator in a variety of graph formats. It is expected each organization will develop its own format for graphics. Underlying the information in the graphs, an organization will need to investigate root causes and related information to fully understand what is being flagged by the indicator.*

The format of the leading indicators information has been developed to be consistent with widely accepted measurement guidance in use in systems engineering and software organizations to include the references listed in Section 5. In an attempt to normalize this guidance, consider the following depiction of a leading indicator.



A leading indicator is composed of characteristics, a condition and a predicted behavior. The characteristics are detailed in a table contained in Appendix F. That table describes the typical anatomy of the information measurement specification used throughout this guide. The characteristics and condition are analyzed on a periodic or as-needed basis. The leading indicator predicts behavior within a given confidence and within an accepted time range into the future. Based on accuracy within an organization, a leading indicator is given a strength value that represents the belief an organization has in the indicator. As organizational experience proves the value of the leading indicator the organization would increase the strength value.

For example, an invalidated leading indicator that is used for the first time might predict that there is a 100% probability that the project will fail in 3 months. However, managers would not rely heavily on this leading indicator because it has a very low strength, in this case 0.

Within regard to base measures, an important assumption and implementation consideration is the additional associated attributes are also recorded with the base measurement values. These attributes are necessary to aid in categorization and analysis. Attributes are needed to convert the Base Measures into Derived Measures. This additional insight makes indicator data much more relevant to the information users. Without such data, users would find it difficult, if not impossible to appropriately interpret the indicator or to investigate, and potentially, take appropriate corrective action.

These categories should be selected by relevance to the organizational. As examples, some very useful attribute categories to consider include ISO/IEC 15288, IEEE 1220 and ISO/IEC 26702 Process Milestone, Stage, or Phases, Disposition Action (opened, started, approved, incorporated, rejected, stopped, closed, overdue), Maturity States (planned, interim, actual), Priority Levels (critical, high, medium, low), Cause (error, customer request, external), Impact Level (high, medium, low), Classification Type, and Dates & Times of Associated Events. This list of attributes is not exhaustive. It is simply a list of those attributes widely used in the existing leading indicators.

For example, "Requirements" must be converted to "% Requirements Approved" ('Approved' being a Disposition Action) for the user to get a sense of the system definition progress and maturity. Further, if at a monthly status review the "% Requirements Approved" is observed to be trending lower than established thresholds, then appropriate action can be taken. Ideally, additional associated attributes are also recorded, including but not limited to the Process Phases, Disposition Action, Maturity States, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times associate with the requirement approval events.

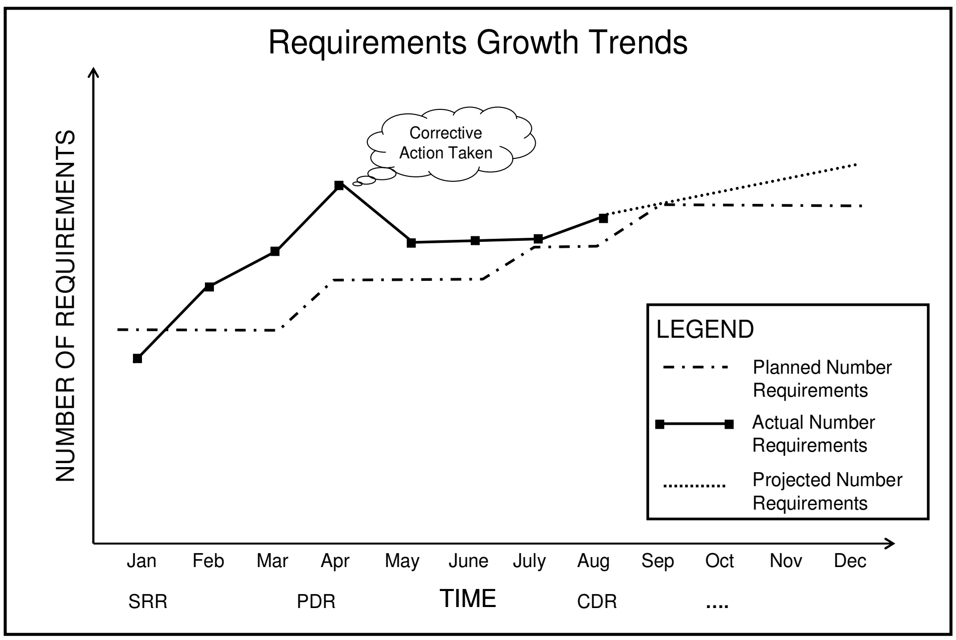
Further uses of attributes are presented in the various analysis guidance sections of the leading indicator specifications.

3.1 Requirements Trends

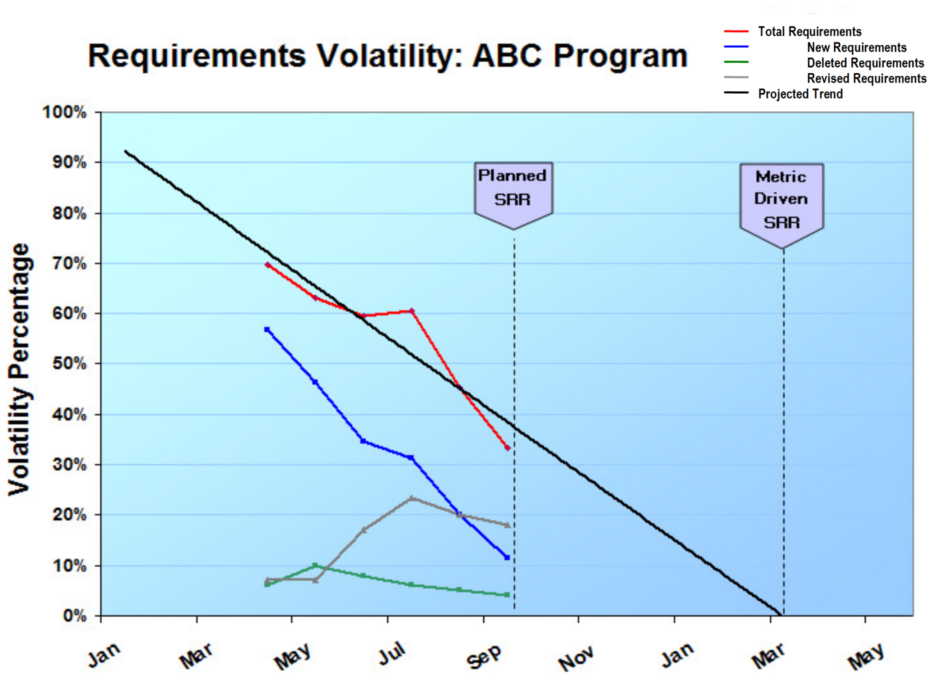
This indicator is used to evaluate the trends in the growth, change, completeness and correctness of the definition of the system requirements. This indicator provides insight into the rate of maturity of the system definition against the plan. Additionally, it characterizes the stability and completeness of the system requirements which could potentially impact design, production, operational utility, or support. The interface trends can also indicate risks of change to and quality of architecture, design, implementation, verification, and validation, as well as potential impact to cost and schedule.

An example of how such an indicator might be reported is show below. Refer to the measurement information specification below for the details regarding this indicator; the specification includes the general information which would be tailored by each organization to suit its needs and organizational practices.

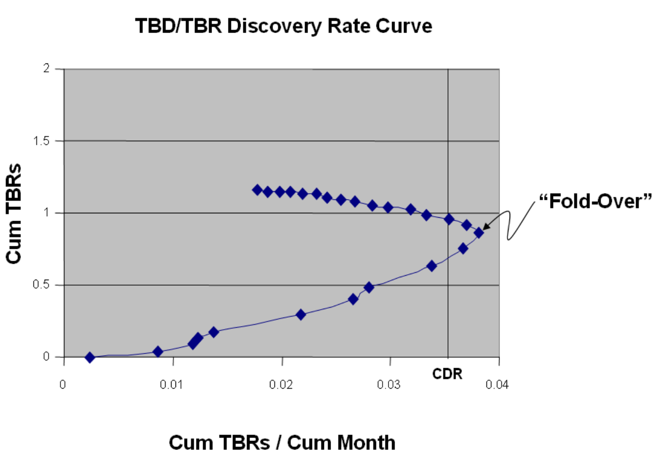
Requirements Trends

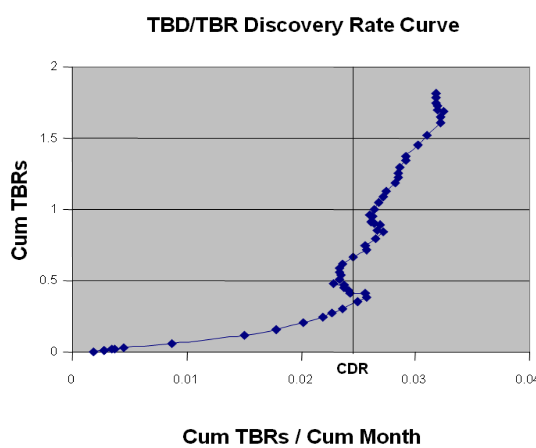


Requirements Trends. The graph illustrates growth trends in the total number of active requirements in respect to planned number of requirements (which is typically based on expected value based on historical information of similar projects as well as the nature of the project). The measures shown could apply to all levels of abstraction from high-level to detailed requirements. Based on actual data, a projected number of requirements will also be shown on a graph. In this case, we can see around PDR that there is a significant variance in actual versus planned requirements, indicating a growing problem. An organization would then take corrective action – where we would expect to see the actual growth move back toward the planned subsequent to this point. The requirements growth is an indicator of potential impacts to cost, schedule, and complexity of the technical solution. It also indicates risks of change to and quality of architecture, design, implementation, verification, and validation.



Requirements Volatility. The graph illustrates the rate of change of requirements over time. It also provides a profile of the types of change (new, deleted, or revised) which allows root-cause analysis of the change drivers. By monitoring the requirements volatility trend, the project team is able to predict the readiness for the System Requirements Review (SRR) milestone. In this example, the project team initially selected a calendar date to conduct the SRR, but in subsequent planning made the decision to have the SRR be event driven, resulting in a new date for the review wherein there could be a successful review outcome.





TBD/TBR Discovery Rate. The graphs show the cumulative requirement TBDs/TBRs vs. the ratio of cumulative TBDs/TBRs over cumulative time. Each point represents a successive instance in time as you move along the graph from bottom to top. The plot provides an indication of the convergence and stability of the TBDs/TBRs over the life cycle of the project. The graph on the left shows a desirable trend of requirement TBD/TBR stability; as the ratio of decreases and the cumulative number of TBDs/TBRs approaches a constant level. This "fold-over" pattern is the desirable trend to look for, especially in the later stages of project life cycle. In contrast, the graph on the right shows an increasing number of

TBDs/TBRs even as the project approaches later stages of its life cycle; this is a worrisome trend in system design stability. An advantage of this plot is that, by shape of the graph (without having to read into the quantitative values), one can get a definitive idea for the trend of requirement stability, particularly when it is associated with certain key project milestones. The similar graphing technique can be applied to TBDs/TBRs in the Interface Trends indicator and to the number of Requests for Change (RFCs) measure for the System Definition Change Backlog Trends indicator.

3.1.1 Requirements Trend Specification

Requirements Trends	
Information Need Description	
Information Need	• Evaluate the stability and adequacy of the requirements to understand the risks to other activities towards providing required capability, on-time and within budget. • Understand the growth, change, completeness and correctness of the definition of the system requirements.
Information Category	1. Product size and stability – Functional Size and Stability 2. Also may relate to Product Quality and Process Performance (relative to effectiveness and efficiency of validation)
Measurable Concept and Leading Insight	
Measurable Concept	Is the SE effort driving towards stability in the System definition and size?
Leading Insight Provided	• Indicates whether the system definition is maturing as expected. • Indicates risks of change to and quality of architecture, design, implementation, verification, and validation. • Indicates schedule and cost risks. • Greater requirements growth, changes, or impacts than planned or lower closure rate of TBDs/TBRs than planned indicate these risks. • May indicate future need for different level or type of resources/skills. • Indicates potential lack of understanding of stakeholder requirements that may lead to operational or supportability deficiencies.
Base Measure Specification	
Base Measures	1. Requirements 2. Requirement TBDs/TBRs 3. Requirement Defects 4. Requirement Changes 5. Requirement Change Impact
Measurement Methods	1. Count the number of Requirements (record associated attributes of interest; e.g., Process Phases, Disposition Action, Maturity States, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times) 2. Count the number of Requirement TBDs/TBRs (record associated attributes of interest; e.g., Process Phases, Disposition Action, Maturity States, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times) 3. Count the number of Requirement Defects (record associated attributes of interest; e.g., Process Phases, Disposition Action, Maturity States, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times) 4. Count the number of Requirement Changes (record associated attributes of interest; e.g., Process Phases, Disposition Action, Maturity States, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times) 5. Estimate the impact of a Requirement Change

Requirements Trends	
Unit of Measurement	1. Requirements 2. Requirement TBDs/TBRs per associated attributes 3. Requirement Defects per associated attributes 4. Requirement Changes per associated attributes 5. Effort Hours per Requirement Change (effort hours or range of effort hours expected for each change)
Entities and Attributes	
Relevant Entities	• Requirements
Attributes	• Requirement TBDs/TBRs • Requirement Defects • Requirement Changes • Additional attributes including but not limited to the Process Phases, Disposition Action, Maturity States, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times coupled with the associated events
Derived Measure Specification	
Derived Measure	1. % Requirements Approved 2. % Requirements Growth 3. % TBDs/TBRs Closure Variance per Plan 4. % Requirements Modified 5. Estimated Impact of Requirements Changes for a given time interval (in Effort Hours) 6. Requirement Defect Profile 7. Requirement Defect Density 8. Requirement Defect Leakage (or Escapes) 9. Cycle time for Requirement Changes (each and average)
Measurement Function *	1. (Requirements Approved / Requirements identified and defined)*100 for a given time interval 2. ((Requirements in current baseline - Requirements in previous baseline) / (Requirements in previous baseline) * 100 3. ((TBDs/TBRs planned for closure – TBDs/TBRs closed) / TBDs/TBRs planned for closure) * 100 4. (Requirements Modified / Total Requirements) * 100 for a given time interval 5. Sum of estimated impacts of Requirement Changes during a given time interval 6. Requirement Defects for each defect category 7. Requirement Defects / Requirements as a function of time 8. Subset of Requirement Defects found in a phase subsequent to its insertion 9. Elapsed time (difference between start and stop times) or total effort hours for each Requirements Change

Requirements Trends	
Indicator Specification	
Indicator Description and Sample	Line or bar graphs that show trends of requirements growth and TBD/TBR closure per plan. Stacked bar graph that shows types, causes, and impact/severity of changes. Show thresholds of expected values based on experiential data. Show key events along the time axis of the graphs. 1. Line or bar graphs that show growth of Requirements over time 2. Line or bar graphs that show % Requirements Approved over time 3. Line or bar graphs that show % TBDs/TBRs not closed per plan 4. Line or bar graphs that show % Requirements Change 5. Line or bar graphs that show Estimated Impact of Requirements Change for a given time interval (in effort hours) 6. Line or bar graphs that show Defect Profile (by types, causes, severity, etc.) 7. Line or bar graphs that show Defect Density 8. Stacked bar graph that shows types, causes, and impact/severity of Requirements Changes
Thresholds and Outliers	Organization dependent.
Decision Criteria	Investigate, and potentially, take corrective action when the requirements growth, requirements change impact, or defect density/distribution exceeds established thresholds <fill in organization specific threshold> or a trend is observed per established guidelines <fill in organizational specific>.

Indicator Interpretation	• Used to understand the maturity of the system definition
	• Used to understand impact on system definition and impact on production.
	• Analyze this indicator for process performance and other relationships that may provide more "leading perspective".
	• Ops Concept quality may be a significant leading indicator of the requirements stability (may be able to use number of review comments; stakeholder coverage in defining the Ops Concept).
	• Care should be taken that the organization does not create incentives driving perceptions that all requirements change is undesirable. Note: Requirements changes may be necessary to accommodate new functionality.
	• Review of this indicator can help determine the adequacy of:
	  o Quantity and quality of Systems Engineers
	  o Infrastructure
	  o Process maturity (acquirer and supplier)
	  o Interface design capability
	  o Stakeholder collaboration across life cycle
	• Funding by customer; financial challenge by the program management

Requirements Trends	
Additional Information	
Related Processes	Stakeholder Requirements, Requirements Analysis, Architectural Design
Assumptions	• Requirements Database, Change Control records, defect records are maintained & current. • TBDs and TBRs are recorded and tracked.
Additional Analysis Guidance	• May also be helpful to track trends based on severity/priority of changes • Defect leakage - identify the phases in which defect was inserted and found for each defect recorded.

Implementation Considerations	• Requirements that are not at least at the point of a draft baseline should not be counted.
	• Usage is driven by the correctness and stability of requirements definition.
	  ○ Lower stability means higher risk of impact to other activities and other phases, thus requiring more frequent review.
	  ○ Applies throughout the life cycle, based on risk.
	  ○ Track this information per baseline version to track the maturity of the baseline as the system definition evolves.

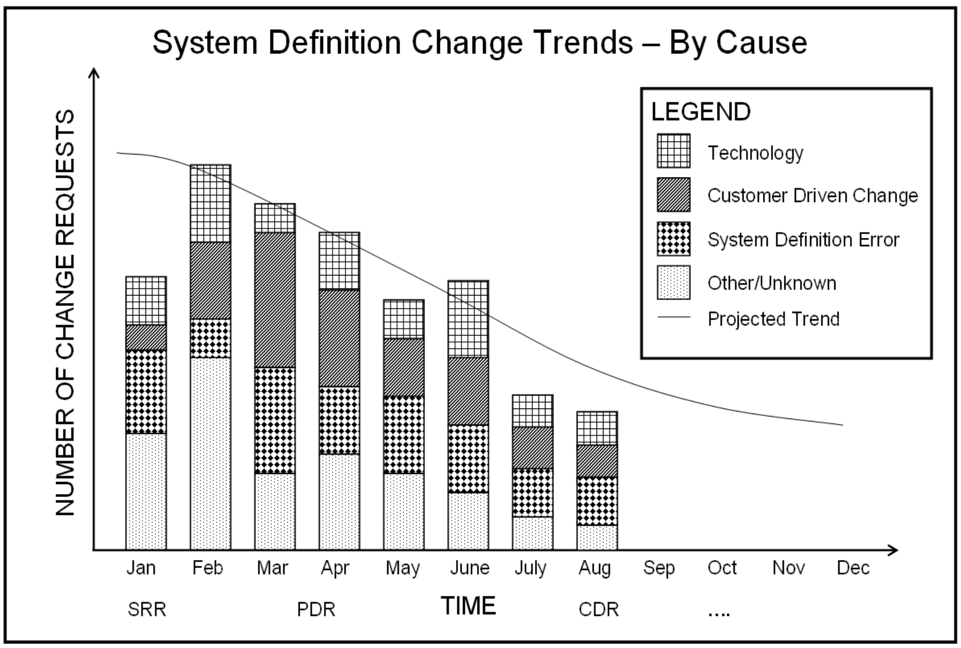
User of Information	• Program/Project Manager • Chief Systems Engineer • Product Managers • Designers
Data Collection Procedure	• See Appendix F
Data Analysis Procedure	• See Appendix F

3.2 System Definition Change Backlog Trends

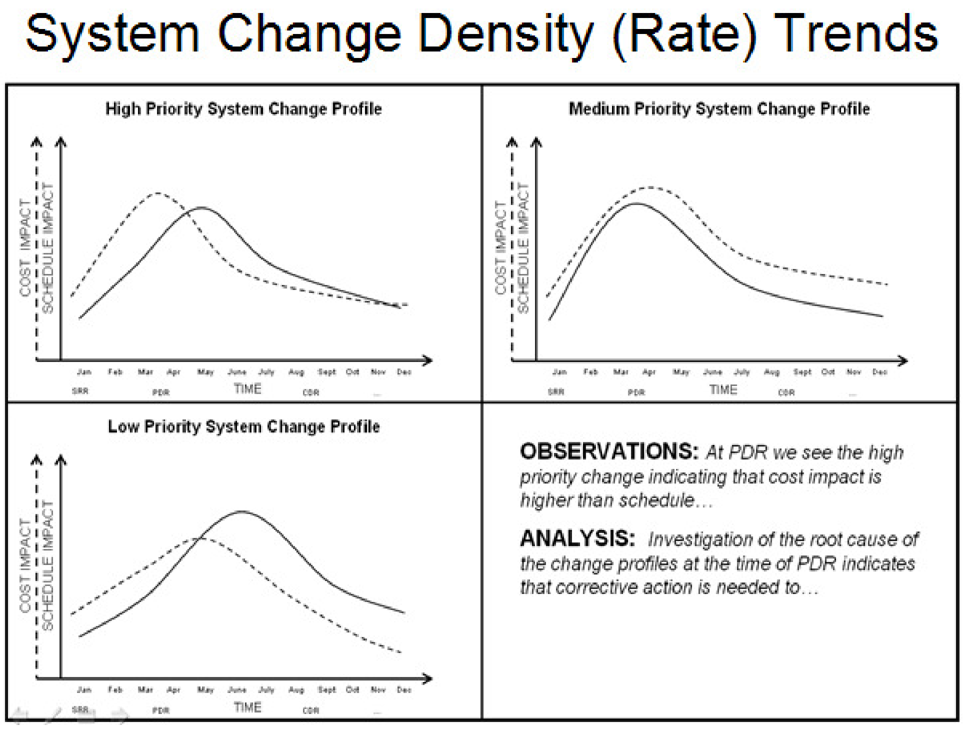
This indicator is used to evaluate the trends in system definition change backlog, indicating whether the change backlog is impeding system definition progress or system development quality/schedule. It may also provide an indication of potential rework due to changes not being available in a timely manner.

An example of how such an indicator might be reported is show below. Refer to the measurement information specification below for the details regarding this indicator; the specification includes the general information which would be tailored by each organization to suit its needs and organizational practices.

System Definition Change Backlog Trends

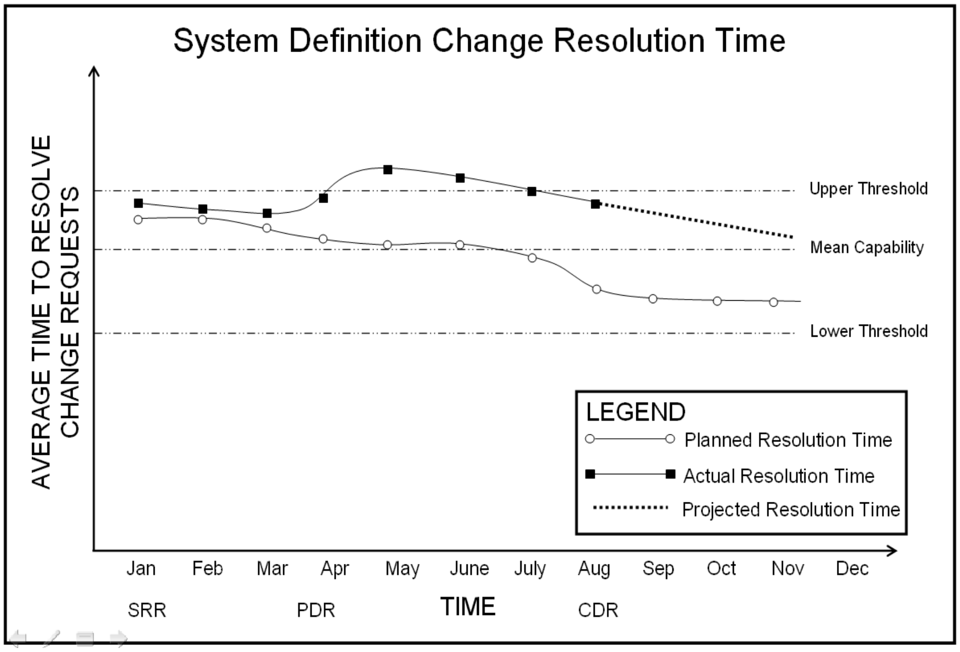


System Definition Change Backlog Trends. The graphs included here illustrate the system definition change trend with respect to the historically based expected trend of changes. In the case of the System Definition Change Trend – By Cause example, we see at SRR there are actually less changes than expected, and the project might need to investigate the factors for this to determine if this is a concern, and perhaps may lead to higher levels of change later in the project. The number of Change Requests in the month following the SRR, could project to a very challenging trend, but generally falls within historical experience. Fortunately, the trend observed between the SRR and the PDR tracks remains in line with historical experience, perhaps suggesting that no significant issues exist with respect to the total number of changes. The organization may find it useful investigate the individual trends associated with the changes categorized according to cause. A very mature organization might have expected trend lines for each type of change.

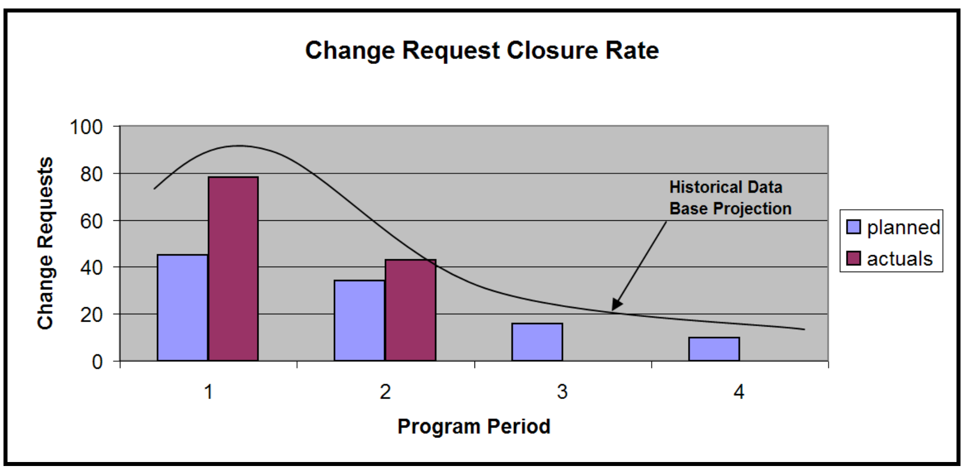


System Change Density (Rate) Trend. Given that Cost and Schedule are normalized on the vertical axis graphs above, these System Change Density (Rate) Trend examples, illustrate this indicator is used to evaluate the changes categorized according to priority over time in terms of cost and schedule impact. It indicates whether the project is effectively managing the project changes as shown by predicted impact ratings over time. If the impacts continue to grow or not be reduced, the customer satisfaction may be negatively impacted due to resulting cost, schedule, or technical impacts. In addition to the change data itself, the average time to resolve the change requests provides additional leading information, as shown in the example graphs below.

System Definition Change Trends



System Definition Change Resolution Time. The graph illustrates the average time to resolve change requests versus what is planned for the project or historical data. Based on historical data and nature of the project, a projection is made for the future; In this case, the actual data depicted through Program Period 2 warrants further analysis as it is significantly over the expectations (it is neither to project plan or historical-based projects) and may not be trending appropriately over time. Mature organizations should be able to identify lower and upper thresholds, as well as average time (organization's mean capability), to resolve a change.



Change Request Closure Rate. The graph illustrates the number of change requests resolved versus what is planned for the project based on historical data and nature of the project. Based on actual data to date, a projection is made for the future. The graph used for the Requirement TBD/TBR Discovery Rate can also be applied to plot the Request for Changes (RFCs) to indicate the trend for system definition and design stability.

3.2.1 System Definition Change Backlog Trend Specification

System Definition Change Backlog Trends	
Information Need Description	
Information Need	Evaluate the backlog trends of the system definition to understand whether the changes are being made in a timely manner
Information Category	1. Schedule and Progress – Work Unit Progress 2. Also may relate to Process Performance - Process Efficiency 3. Also may relate to Product Stability
Measurable Concept and Leading Insight	
Measurable Concept	Are changes to the baseline being processed in a systematic and timely manner?
Leading Insight Provided	• Indicates whether the change backlog is impeding system definition progress or system development quality/schedule. • Indication of potential rework due to changes not being available in a timely manner.
Base Measure Specification	
Base Measures	1. Requests For Change 2. Duration of Change 3. Changes by Priority 4. Changes by Cause 5. Changes by Disposition Action 6. Impact of Change
Measurement Methods	1. Count the number of Requests For Changes 2. Record Change Events (record associated attributes of interest; actual dates & times from CM system) 3-5. Count the number of Changes (record associated attributes of interest; e.g. Priority Levels, Cause, Disposition Action) 6. Estimate (in estimated effort or cost) based on engineering judgment and documented in the Change Request.
Unit of Measurement	1. Requests For Changes 2. Days, Hours, or Minutes 3. Changes by Priority 4. Changes by Cause 5. Changes by Disposition 6. Assessed qualitative impact or Effort (Days, Hours, Minutes) or Cost

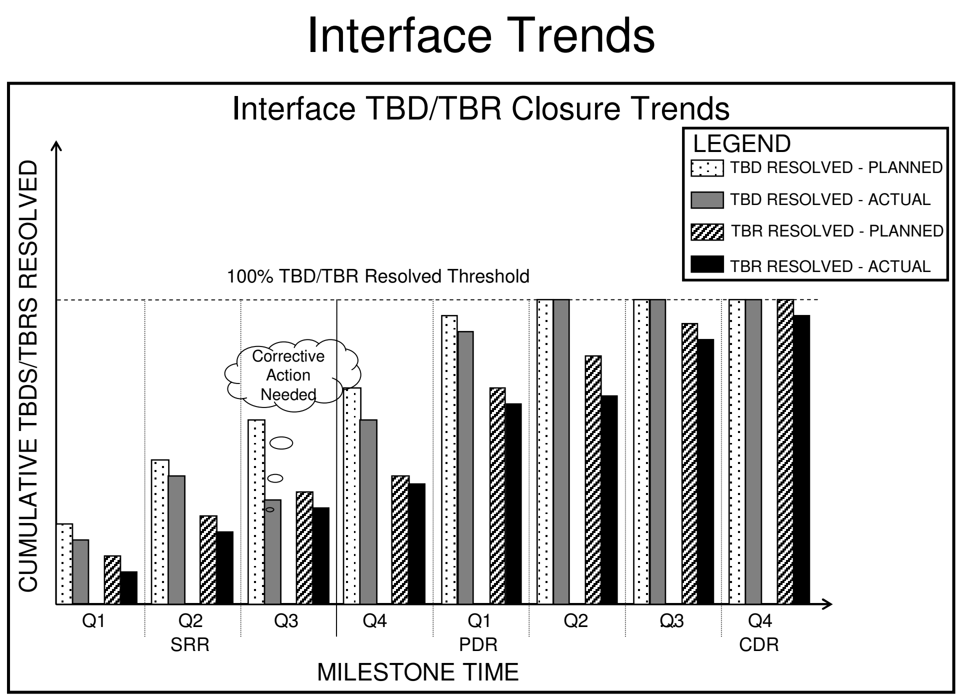
System Definition Change Backlog Trends	
Entities and Attributes	
Relevant Entities	• Request(s) For Change (RFCs)
Attributes	• Requirement Changes • Additional attributes including but not limited to the Process Phases, Disposition Action, Maturity States, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times coupled with the associated events
Derived Measure Specification	
Derived Measure	1. Approval/Closure Rates 2. Cycle Time Statistical Measures by Associated attributes (e.g., mean, mode, min/max, dev.) 3. Priority Density
Measurement Function	1. (RFCs approved / RFCs submitted) * 100 [per time interval] 2. Time approved – Time submitted (record associated attributes of interest) 3. (RFCs by priority / RFCs)
Indicator Specification	
Indicator Description and Sample	• Line graphs that show trends of RFC cycle time and backlog status over time. • Pareto graph or stacked bar graph that shows types, causes, and impact/severity of changes. • Line graphs that show projections of when the current backlog will be closed (using rate of arrivals, plus rate of closure) • Show thresholds of expected values based on experiential data.
Thresholds and Outliers	User defined.
Decision Criteria	Investigate and, potentially, take corrective action when the change backlog exceeds established thresholds <fill in organization specific threshold> or a trend is observed per established guidelines <fill in organizational specific>.
Indicator Interpretation	Used to understand impact on system definition and development progress, and impact on time to market, and to identify associated risks. Also to provide insight to the level of capacity required to correctly process a change (resources, skill set).

System Definition Change Backlog Trends	
Additional Information	
Related Processes	Stakeholder requirements, Requirements Analysis, Architectural Design, Requirements Management
Assumptions	• Requirements Database and Change Control records are maintained & current. • Interface requirements are included in the requirements database. • TBDs and TBRs are recorded and tracked.
Additional Analysis Guidance	• Also provides useful lagging information: Indicates that the SE processes are not being implemented effectively. • Are people reviewing the system definition at the appropriate level
Implementation Considerations	• Use whenever there are multiple changes in the approval queue, after baseline has been established. More frequent review needed when backlog increases, especially if changes have interdependencies. • Do not sample - collect all RFC data. • Analyze this indicator for other relationships that may provide more "leading perspective". • Relationship between open/unresolved changes needs to be considered. • Review the results of this indicator against rework to see if growing backlog is resulting in more rework.
User of Information	• Program/Project Manager – associated risks affecting program and project execution, level of capacity required • Chief Systems Engineer – impact of system definition and development activity, level of capacity required • Configuration Management Manager – process indicator
Data Collection Procedure	• See Appendix F
Data Analysis Procedure	• See Appendix F

3.3 Interface Trends

This indicator is used to evaluate the trends related to growth, change, completeness, and correctness of the definition of system interfaces. This indicator provides insight into the rate of maturity of the system definition against the plan. It also assists in helping to evaluate the stability and adequacy of the interfaces to understand the risks to other activities towards providing required capability, on-time and within budget. The interface trends can also indicate risks of change to and quality of architecture, design, implementation, verification, and validation, as well as potential impact to cost and schedule.

An example of how such an indicator might be reported is show below. Refer to the measurement information specification below for the details regarding this indicator; the specification includes the general information which would be tailored by each organization to suit its needs and organizational practice.



Interface TBD/TBR Closure Trends. The graph illustrates the actual cumulative number of TBDs and TBRs that have been resolved compared to what is planned to be resolved based on historical data and expectations given the project characteristics. It can be seen that in Q3 after SRR, the actual TBDs are significantly lower than planned and corrective action is then taken.

The graph used for the Requirement TBD/TBR Discovery Rate can also be applied to plot the Interface TBD/TBR trends to indicate the trend for system interface definition and design stability.

3.3.1 Interface Trend Specification

Interface Trends	
	Information Need Description
Information Need	• Evaluate the stability and adequacy of the interfaces to understand the risks to other activities towards providing required capability, on-time and within budget. • Understand the growth, change, and correctness of the definition of the system interfaces.
Information Category	1. Product size and stability – Functional Size and Stability 2. Also may relate to Product Quality and Process Performance (relative to effectiveness and efficiency of validation)
	Measurable Concept and Leading Insight
Measurable Concept	Is the SE effort driving towards correctness and completeness (i.e., approved) of the definition and design of interfaces?
Leading Insight Provided	• Indicates whether the system definition is maturing as expected. Unfavorable trends indicate high risk during design, implementation and/or integration. • Indicates risks of change to and quality of architecture, design, implementation, verification, and validation. • Greater interface growth, changes, or impacts than planned or lower closure rate of TBDs/TBRs than planned indicate risks to the system definition and flow-down. • May indicate future need for different level or type of resources/skills.
	Base Measure Specification
Base Measures	1. Interface(s) 2. Interface TBDs/TBRs 3. Interface Defects 4. Interface Changes 5. Interface Change Impact
Measurement Methods	1. Count the number of Interface(s) (record associated attributes of interest; e.g., status, maturity - identified and defined) 2. Count the number of Interface TBDs/TBRs (record associated attributes of interest) among those interfaces identified and defined 3. Count the number of Interface Defects (record associated attributes of interest; e.g., type, cause, severity) 4. Count the number of Interface Changes (record associated attributes of interest; e.g., type, cause) 5. Estimate the Effort Hours per Interface Change
Unit of Measurement	1. Interfaces 2. Interface TBDs/TBRs 3. Interface Defects 4. Interface Changes 5. Effort hours per Interface Change (effort hours or range of effort expected for each change)

Interface Trends	
Entities and Attributes	
Relevant Entities	• Interfaces
Attributes	• Interface TBDs/TBRs • Interface Defects • Interface Changes • Interface Change Impact • Additional attributes including but not limited to the Process Phases, Disposition Action, Maturity States, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times coupled with the associated events
Derived Measure Specification	
Derived Measure	1. % Interfaces approved 2. % Interfaces growth 3. TBDs/TBRs closure variance per plan 4. % Interfaces modified 5. Estimated Impact of Changes for time interval (in effort hours), 6. Defect profile 7. Defect density 8. Defect leakage (or escapes) 9. Cycle time for interface changes (each and average) 10. Rate of convergence of interfaces
Measurement Function	1. (Interfaces Approved / Interfaces identified and defined)* 100 for a given time interval 2. ((Interfaces in current baseline - Interfaces in previous baseline) / (Interfaces in previous baseline) * 100 3. ((TBDs/TBRs planned for closure – TBDs/TBRs closed) / TBDs/TBRs planned for closure) * 100 4. (Interfaces Modified / Total Interfaces) * 100 for a given time interval 5. Sum of estimated impacts of Interfaces Changes during a given time interval 6. Interface Defects for each defect category 7. Interface Defects / Interface as a function of time 8. Subset of Interface Defects found in a phase subsequent to its insertion 9. Elapsed time (difference between completion time and start times) or total effort hours for each Interface Defects 10. Interface as a function of time

Interface Trends	
Indicator Specification	
Indicator Description and Sample	Line or bar graphs that show trends of interface approval rates and TBD/TBR closure per plan. Stacked bar graph that shows types, causes, and impact/severity of changes. Show thresholds of expected values based on experiential data. Show key events along the time axis of the graphs. 1. Line or bar graphs that show growth of interfaces over time 2. Line or bar graphs that show % interfaces approved over time 3. Line or bar graphs that show % TBDs/TBRs not closed per plan 4. Line or bar graphs that show % interfaces modified, 5. Line or bar graphs that show estimated impact of changes for time interval (in effort hours) 6. Line or bar graphs that show defect profile (by types, causes, severity, etc.) 7. Line or bar graphs that show defect density 8. Stacked bar graph that shows types, causes, and impact/severity of changes on system design
Thresholds and Outliers	Organization dependent.
Decision Criteria	Investigate and, potentially, take corrective action when the interfaces are faulty and incomplete, interfaces change impact, or defect density/distribution exceeds established thresholds <fill in organization specific threshold> or a trend is observed per established guidelines <fill in organizational specific>.
Indicator Interpretation	• Used to understand impact on system definition, design, and system integration. • Analyze this indicator for process and system definition performance and progress, and impact to architecture, design, implementation, verification, and validation (which may provide more leading "perspective"). • Unfavorable trends indicate high risk during design, implementation and/or integration. • Care should be taken that the organization does not create incentives driving perceptions that all interface changes are undesirable. • Review of this indicator can help determine the adequacy of: o Quantity and quality of Systems Engineers o Infrastructure o Process maturity (acquirer and supplier) o Interface design capability o Stakeholder collaboration across life cycle o Funding by customer; financial challenge by the program management

Interface Trends	
Additional Information	
Related Processes	Stakeholder requirements, Requirements Analysis, Architectural Design
Assumptions	Requirements database, change control records, and defect records are maintained and current.
Additional Analysis Guidance	• May also be helpful to track trends based on severity/priority of changes • Defect leakage – identify the phases in which the defect was inserted and found for each defect recorded.

Implementation Considerations	• Usage is driven by the correctness and stability of interfaces definition and design.
	  ○ Lower stability means higher risk of impact to other activities and other phases, thus requiring more frequent review.
	  ○ Applies throughout the life cycle, based on risk.
	  ○ Track this information per baseline version to track the maturity of the baseline as the system definition evolves.

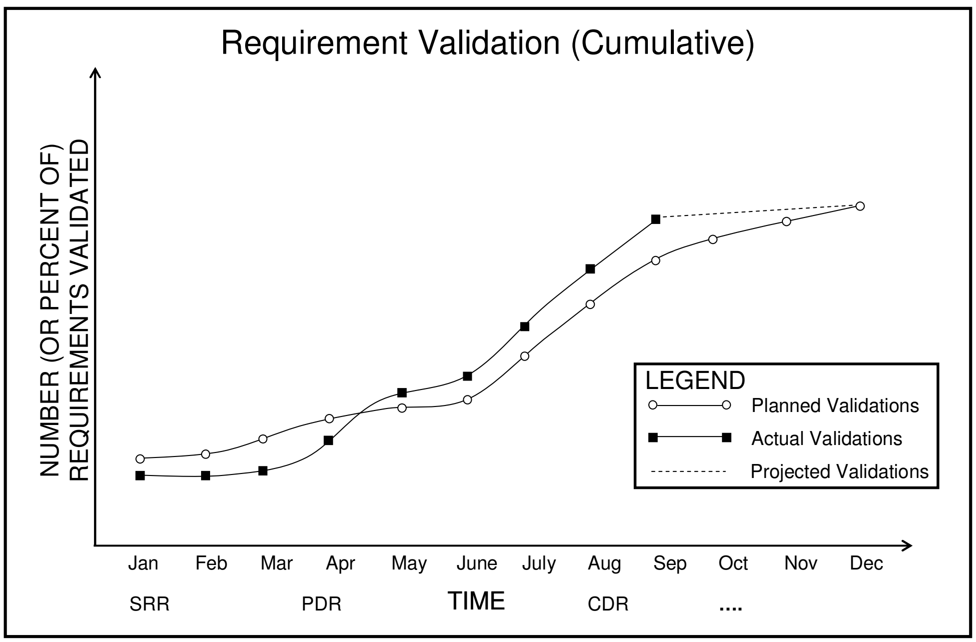
User of Information	• Program/Project Manager • Chief Systems Engineer • Interface Managers • Designers
Data Collection Procedure	• See Appendix F
Data Analysis Procedure	• See Appendix F

3.4 Requirements Validation Trends

This indicator is used to evaluate the trends in the rate and progress of requirements validation activity. It provides early insight into the level of understanding of customer/user needs. It indicates risk to system definition due to inadequate understanding of the customer/user needs. It may also indicate risk of schedule/cost overruns, post delivery changes, operational utility deficiencies, or user dissatisfaction

An example of how such an indicator might be reported is show below. Refer to the measurement information specification below for the details regarding this indicator; the specification includes the general information which would be tailored by each organization to suit its needs and organizational practices.

Requirements Validation Trends



Requirements Validation Trends. The graph illustrates the actual number of (or it could also be shown as the percent of) requirements validated versus the planned validation based on historical data and the nature of the project. A projection will also be made based on the actual validation trend. In this case, we see at CDR that the actual validated requirements in higher than planned, indicating that the validation activity is on track.

3.4.1 Requirements Validation Trend Specification

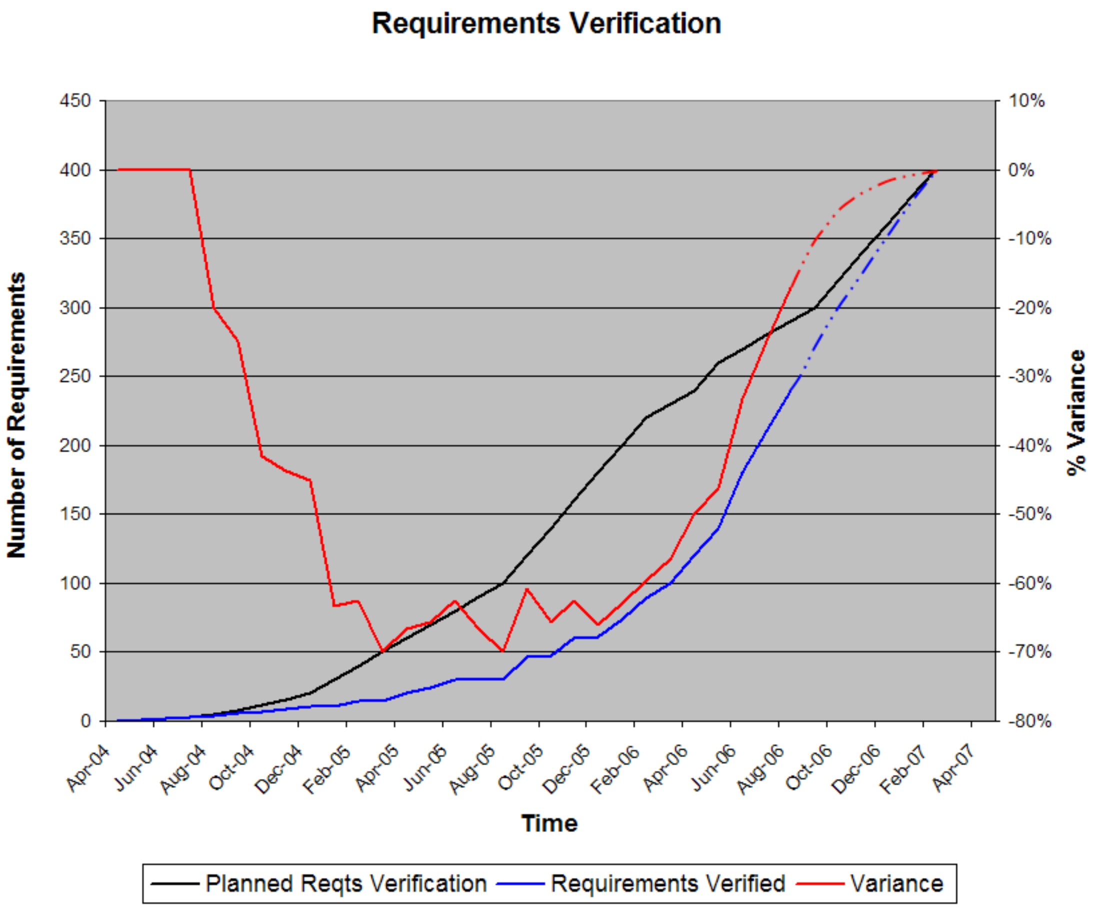
Requirements Validation Rate Trends	
Information Need Description	
Information Need	Understand whether requirements are being validated with the applicable stakeholders at each level of the system development.
Information Category	1. Product Size and Stability – Functional Size and Stability 2. Also may relate to Product Quality and Process Performance (relative to effectiveness and efficiency of validation)
Measurable Concept and Leading Insight	
Measurable Concept	The rate and progress of requirements validation.
Leading Insight Provided	Provides early insight into level of understanding of customer/user needs: • Indicates risk to system definition due to inadequate understanding of the customer/user needs • Indicates risk of schedule/cost overruns, post delivery changes, operational deficiencies, or user dissatisfaction
Base Measure Specification	
Base Measures	1. Requirements 2. Requirements Validated
Measurement Methods	1. Count the number of Requirements 2. Record the number of Requirements planned for Validation (record associated attributes of interest; e.g., time interval, process phase) 3. Count the number of Requirements actually Validated (record associated attributes of interest; e.g., time interval, process phase) 4. Record Requirement Validation Events
Unit of Measurement	1-3.Requirements 4. Date and Time
Entities and Attributes	
Relevant Entities	• Requirements
Attributes	• Maturity State • Stakeholders • Architecture Level • Additional attributes including but not limited to the Process Phases, Disposition Action, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times coupled with the associated events
Derived Measure Specification	
Derived Measure	1. Requirement Validation Rate (Rate at which requirements are validated with the customer/end user) 2. % requirements validated
Measurement Function	1. (Requirements Validated/unit time) 2. (Requirements Validated/Total Requirements)*100

Requirements Validation Rate Trends	
Indicator Specification	
Indicator Description and Sample	1. Line graphs that show trends of validation rates per plan during a validation activity. 2. Table or graph showing time interval or events versus number or percent requirements validated (actual and planned).
Thresholds and Outliers	Organization dependent. Thresholds are phase dependent.
Decision Criteria	Investigate and potentially take corrective action when the validation rate is lower than the established thresholds <fill in organization specific threshold> or a trend is observed per established guidelines <fill in organizational specific>.
Indicator Interpretation	• Investigation is driven by deviation of actual rate, percentage or quantity from plan. • Lower validation rate compared to plan means higher risk, thus it would be reviewed more frequently. If actual validation (rate) is below planned validation (rate), there may be a need to increase staffing, increase review time with customer/end user, and/or review effectiveness of mission/requirements analysis processes pending causal analysis. This can in turn affect quality of system definition, validation, and customer satisfaction. An additional consideration is to examine whether requirements creep could be a source of the lower validation rate. • If the actual validation rate is exceeding the planned validation rate significantly, there may still be risk to consider. The planning process should be reviewed or the quality of the requirement validation method should be analyzed to ensure adequacy, if no process improvement was the reason for the deviation. If planning uses too low of validation rate, then efficiency may be lost. If validation process does not ensure adequate customer/user review, then there may be surprises during system validation.

Requirements Validation Rate Trends	
Additional Information	
Related Processes	Stakeholder requirements, Requirements Analysis, Architectural Design.
Assumptions	Requirements database is maintained and validation rates can be obtained from project timeline. Assumes that appropriate historical database is available.
Additional Analysis Guidance	The timing for validation may be driven by large project reviews/events such as the PDR or CDR. These should be considered in the planning and analysis.
Implementation Considerations	• Usage is driven by the requirements validation rate. • Applies throughout the life cycle, based on risk, but in some cases it may be back-loaded (to SRR or later). • Could apply any time the project has requirements validation scheduled. • If the requirements validation rate is below plan, then there may further investigation warranted to determine what this issue/root cause is. • May also want to consider using "Requirements Validation Results Trends" that looks at causes of validation rejections, etc.
User of Information	• Chief Systems Engineer • V&V Lead • Program/Project Manager • Customer or Third Party V&V
Data Collection Procedure	• See Appendix F
Data Analysis Procedure	• See Appendix F

3.5 Requirements Verification Trends

This indicator is used to evaluate the trends in the rate and progress of requirements verification. It provides early insight into the ability to meet customer/user requirements. The measure indicates possible risk to system definition due to inadequate ability to meet the customer/user requirements. It may indicate risk of schedule/cost overruns, potential for post delivery post delivery changes, operational utility deficiencies, or customer/user dissatisfaction.



Requirements Verification. The graph above illustrates the number of requirements verified monthly versus the planned verification - based on historical data and the nature of the project. The percentage variance is graphed over the same time period using. In addition, this indicator can be reported using a graph similar to that shown for requirements validation.

3.5.1 Requirements Verification Trend Specification

Requirements Verification Trends	
Information Need Description	
Information Need	Understand whether requirements are being verified relative to plan at each level of the system development.
Information Category	1. Product Size and Stability – Functional Size and Stability 2. Also may relate to Product Quality and Process Performance (relative to effectiveness and efficiency of verification)
Measurable Concept and Leading Insight	
Measurable Concept	The rate and progress of requirements verification.
Leading Insight Provided	Provides early insight into ability to meet customer/user requirements: • Indicates risk to system definition due to inadequate ability to meet the customer/user requirements • Indicates risk of schedule/cost overruns, post delivery changes, operational utility deficiencies, or customer/user dissatisfaction
Base Measure Specification	
Base Measures	1. Requirements 2. Requirements Verified
Measurement Methods	1. Count the number of Requirements 2. Record the number of Requirements planned for Verification (record associated attributes of interest; e.g., time interval, process phase) 3. Count the number of Requirements actually Verified (record associated attributes of interest; e.g., time interval, process phase) 4. Record Requirement Verification Events
Unit of Measurement	1-3.Requirements 4. Date and Time
Entities and Attributes	
Relevant Entities	• Requirements
Attributes	• Maturity State • Stakeholders • Architecture Level • Additional attributes including but not limited to the Process Phases, Disposition Action, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times coupled with the associated events
Derived Measure Specification	
Derived Measure	1. Requirements Verification Rate (Rate at which requirements are verified) 2. % Requirements Verified
Measurement Function	1. (Requirements Verified/unit time) 2. (Requirements Verified/total Requirements)*100

Requirements Verification Trends	
	Indicator Specification
Indicator Description and Sample	1. Line graphs that show trends of verification rates per plan during a verification activity. 2. Table or graph showing time interval or events versus number or percent requirements verified (actual and planned).
Thresholds and Outliers	Organization dependent. Thresholds are phase dependent.
Decision Criteria	Investigate and potentially take corrective action when the verification rate is lower than the established thresholds <fill in organization specific threshold> or a trend is observed per established guidelines <fill in organizational specific>.
Indicator Interpretation	• Investigation is driven by deviation of actual rate, percentage or quantity from plan. • Lower verification rate compared to plan means higher risk, thus it would be reviewed more frequently. If the actual verification (rate) is below planned verification (rate), there may be a need to increase staffing, increase verification time with customer/end user, and/or review effectiveness of mission/requirements analysis processes pending causal analysis. This can in turn affect the quality of the system definition, system validation, and customer satisfaction. Lower verification could indicate a problem with test scheduling. • If the actual verification rate is exceeding the planned verification rate significantly, there may still be risk to consider. The planning process should be reviewed or the quality of the requirement verification method should be analyzed to ensure adequacy, if no process improvement was the reason for the deviation. If planning uses too low of verification rate, then efficiency may be lost.

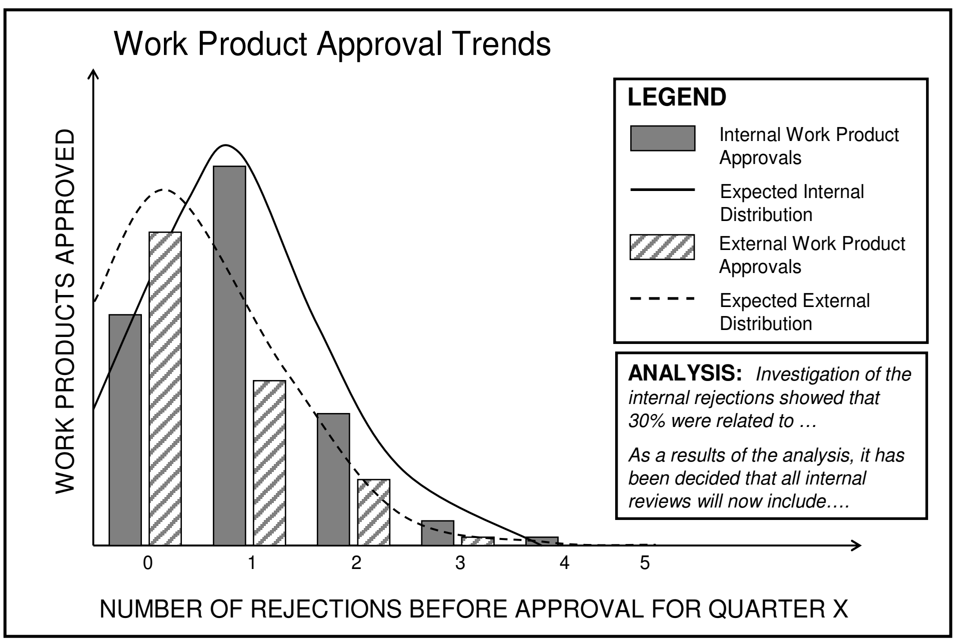
Requirements Verification Trends	
Additional Information	
Related Processes	Stakeholder requirements, Requirements Analysis, Architectural Design.
Assumptions	Requirements database is maintained and verification rates can be obtained from project timeline. Assumes appropriate historical data is available.
Additional Analysis Guidance	The timing for verification may be driven by large project reviews/events such as the PDR or CDR. These should be considered in the planning and analysis.
Implementation Considerations	• Usage is driven by the requirements verification rate. Applies throughout the life cycle, based on risk. Could apply any time the project has requirements verification scheduled. • If the requirements verification rate is below plan, then there may be further investigation warranted to determine what this issue/root cause is. • May also want to consider using "Requirements Verification Results Trends" that looks at causes of verification failures, etc.
User of Information	• Chief Systems Engineer • Verification & Validation Lead • Program/Project Manager • Customer or Third Party V&V
Data Collection Procedure	• See Appendix F
Data Analysis Procedure	• See Appendix F

3.6 Work Product Approval Trends

This indicator is used to evaluate the trends in the internal and external approvals of work products, e.g. a Systems Engineering Management Plan. It may indicate a problem with identification of needs or transformation into requirements/design. It may also indicate that the end product is not of high enough quality and may result in rework or need for changes in plan. It may also be the case that the review process definition or implementation may be inadequate. On the positive side, the measure will indicate readiness for entry into review milestones. This approach could be applied recursively to include subcontractors.

An example of how such an indicator might be reported is show below. Refer to the measurement information specification below for the details regarding this indicator; the specification includes the general information which would be tailored by each organization to suit its needs and organizational practices.

Work Product Approval Trends



Work Product Approval Trends. The graph illustrates success of the work product approvals for Quarter X in respect to how many rejections there were for work products before approval for both internal work product approvals and external work product approvals. Actual rejections are shown with an overlay of the expected internal and external approvals based on historical data and the nature of the project. Analysis will be needed to understand why rejections are happening, and the graphic could include a breakdown of the root causes as stacked bars, for example, rather than just the single bar. Additionally, it may be helpful to use a quad-chart or other graphical presentation techniques to look at performance on related work products together.

3.6.1 Work Product Approval Trend Specification

Work Product Approval Trend	
Information Need Description	
Information Need	Evaluate work product progress to plan and the approval efficiency of the work products.
Information Category	1. Schedule and Progress – Work unit progress 2. Product Quality 3. Process Performance – Process efficiency
Measurable Concept and Leading Insight	
Measurable Concept	Are the system definition work products being approved as planned?
Leading Insight Provided	• Indicates that there may be a problem with identification of needs or transformation into requirements/design. • Indicates that the end product is not of high enough quality/maturity and may result in rework or need for changes in plan. • Indicates that the review process definition or implementation may be inadequate. • Indicates readiness for entry into review milestones • Early indication of where too much emphasis may be placed on quantity at the expense of quality (process breakdown or gaming the system)
Base Measure Specification	
Base Measures	1. Number of Work Products 2. Number of submitted Work Products
Measurement Methods	1. Count the number of Work Products 2. Count the number of Work Products (record associated attributes of interest; e.g., Classification Type, Disposition Action)
Unit of Measurement	1. Work Products 2. Work Products
Entities and Attributes	
Relevant Entities	• Work Products
Attributes	• Additional attributes including but not limited to the Process Phases, Disposition Action, Maturity States, Classification Type, and Dates & Times coupled with the associated events
Derived Measure Specification	
Derived Measure	1. Approval Rate 2. Distribution of Dispositions, 3. Approval Rate performance
Measurement Function	1. (Number of Work Products approved on first submittal) / (Number Work Products submitted) 2. Number of Work Products rejected before Approval 3. (Actual Work Products Approval Rate) / (Planned Work Products Approval Rate)

Work Product Approval Trend	
Indicator Specification	
Indicator Description and Sample	• Graphs that show trends of approval rates per plan during system definition. • Chart showing approval rate distribution by Work Products type.
Thresholds and Outliers	Organization dependent
Decision Criteria	Investigate and, potentially, take corrective action when the approval rate is lower than established thresholds <fill in organization specific threshold> or a trend is observed per established guidelines <fill in organizational specific>. A positive trend can still indicate a risk or problem exists. E.g., a positive trend can be caused from reviews that are not effective or that there is too much effort being expended on work product preparation and review.
Indicator Interpretation	• Decreasing trends indicate greater risk in the review process or the understanding of user needs. • Increasing trends can indicate risk in thoroughness of reviews or that too much effort is being applied on work product preparation and review. • If external approval rate drops below threshold, it may indicate issue with effectiveness of Engineering Review Board, in-process reviews, and processes supporting product generation • Low external approval rates may also indicate that there is a problem with identification of needs or transformation into requirements/design. Examine together with the requirements and interface trends to see if there is a correlation in the results. • If internal approval rate drops below threshold, it may indicate issue with effectiveness of in-process reviews, and processes supporting product generation • In general, as approval rates drop (both internal and external), it could indicate too much emphasis is placed on quantity at the expense of quality (process breakdown or gaming the system). (If this is happening, this may also indicate that there is no objective standard for the work product.) • If internal approval rate gets close to 100%, it may indicate the internal reviews are not thorough enough. Review results together with the External Approval Rate. If external rate is lower, then the cause is probably the lack of thorough internal reviews. • If external approval rate gets close to 100%, may indicate that too much effort is being expended on KWP preparation and review. • Also can provide insight into adequacy of meeting planned/agreed-to milestones (internal and external). • Can provide insight into one influence of customer satisfaction.

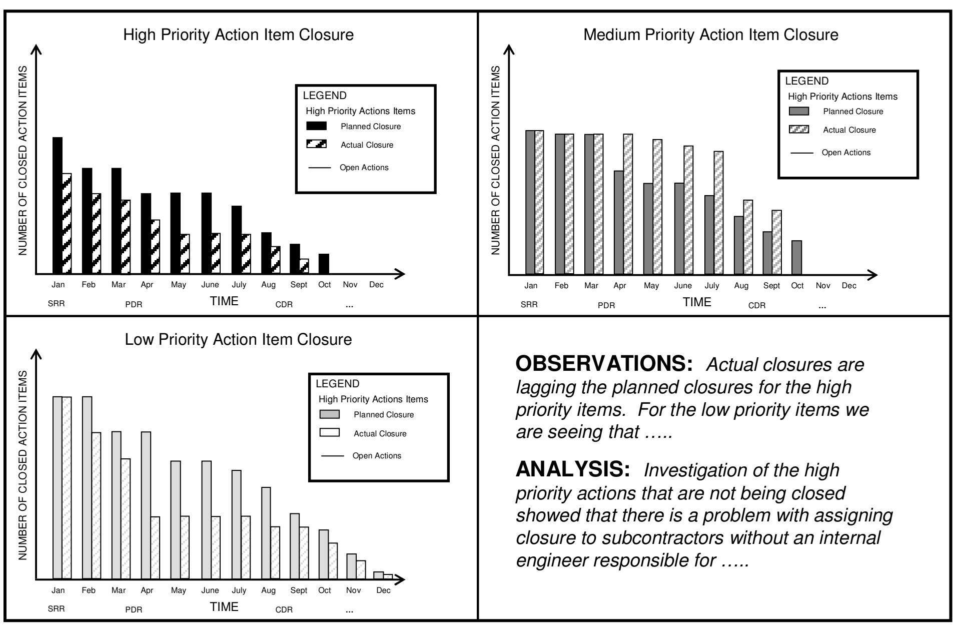
Work Product Approval Trend	
Additional Information	
Related Processes	Review process
Assumptions	• Approval data for work product reviews is captured, retained, and current. • Approval rate based on 1st time submittals. • There is a consistent and validated set of criteria or objective standard for work product review and approval.
Additional Analysis Guidance	• A variation of this indicator is to look at the work product rejection rate. • Could also collect severity of cause of rejections (e.g., major, minor).
Implementation Considerations	• Do not sample - collect all work product approval data. • Use when there are numerous work products going through review and approval. Collect data and use the indicator for both internal (submitted to internal approval authority) and external (submitted to customer approval authority) work product reviews. Not intended for use during interim, incremental, in-process internal reviews. • Most effective if work product review and approval criteria or objective standards are defined, in order to ensure consistent application. • Time interval for data collection and reporting of analysis results may need to change through the life cycle based on phase and level of work product activity.
User Of The Data	• Chief Systems Engineer • Program/Project Manager • Process Owners • Approval Authority
Data Collection Procedure	• See Appendix F
Data Analysis Procedure	• See Appendix F

3.7 Review Action Closure Trends

This indicator is used to evaluate the trends in the closure of review action items. Review actions items may be technical or management/communication related and may be applied to any review or accumulated during related design activities. Large deviations for the planned closure may be indicative of larger, more complex tasks ahead or potentially is a sign of challenging personnel interfaces. In either case, this indicator reveals project risk in terms of rework and/or infeasible schedule. Positive trends will provide insight into readiness to move to the next step/stage/phase. In should be noted that this approach and set of measures could be applied to the management of any set of actions items, not just those associated directly with project reviews.

An example of how such an indicator might be reported is show below. Refer to the measurement information specification below for the details regarding this indicator; the specification includes the general information which would be tailored by each organization to suit its needs and organizational practices.

Review Action Item Closure Trends



Review Action Item Closure Trends. The graph illustrates the number of review action items that are closed in each month, in respect to the number that is planned for closure in that month, based on historical information and nature of the project. The graphic shows the high priority, medium priority, and low priority actions on separate quadrants. A measurement analyst would be able to make observations that would require additional detailed analysis to decide if corrective action was required, and the nature of such action.

3.7.1 Review Action Closure Trend Specification

Review Action Closure Trends	
Information Need Description	
Information Need	Evaluate design review action item progress to plan and closure efficiency.
Information Category	1. Schedule & Progress – Milestone completion 2. Also may relate to Product Quality – efficiency; Process Performance – process efficiency; and Customer Satisfaction – customer feedback
Measurable Concept and Leading Insight	
Measurable Concept	Are early design review action items being closed according to plan?
Leading Insight Provided	• Design review actions items may be technical or management/ communication related. Large deviations for the planned closure may be indicative of larger, more complex tasks ahead or potentially is a sign of challenging personnel interfaces. In either case, this indicator reveals project risk in terms of rework and/or infeasible schedule. • May provide insight into readiness to move to the next step/stage/phase. • May be an indication of the feasibility of the plan with respect to cost, schedule, quality, performance, or functionality.
Base Measure Specification	
Base Measures	1. Action Items 2. Action Items by Priority 3. Action Items by Disposition Action 4. Action Items by Process Phase 5. Impact for Action Item
Measurement Methods	1. Count the number of Action Items 2-4. Count the number of Action Items (record associated attributes of interest; e.g. Priority Levels, Disposition Action, Process Phase event such as Design Review or Milestone and actual dates & times of Dispositions and Events) 5. Estimate (in estimated effort or cost) based on engineering judgment and documented in the Action Item
Unit of Measurement	1. Action Items 2. Action Items by Priority 3. Action Items by Disposition Action 4. Action Items by Process Phase 5. Assessed qualitative impact or Effort (Days, Hours, Minutes) or Cost
Entities and Attributes	
Relevant Entities	• Action Item(s)
Attributes	• Additional attributes including but not limited to the Process Phases, Disposition Action, Priority Levels, Impact Level, Classification Type, and Dates & Times coupled with the associated events
Derived Measure Specification	
Derived Measure	1. Closure rates 2. Action item closure performance 3. Variance from thresholds (for number of action items assigned at design review or closure performance)

<table>
<tr><th colspan="2">Review Action Closure Trends</th></tr>
<tr><td>Measurement Function</td><td>1. Number of action items closed over time
2. (Action items closed over time interval) / (Action items planned for closure over time interval)
3. Difference between observed values and threshold values</td></tr>
<tr><th colspan="2">Indicator Specification</th></tr>
<tr><td>Indicator Description and Sample</td><td>• Graph(s) showing trends of closure rates and action item performance.
• May include bar graph showing total number of actions per review.
• Graphs may show results by priority of actions.
• Show thresholds of expected values based on experiential data.
• Show key events along the time axis of the graph(s).</td></tr>
<tr><td>Thresholds and Outliers</td><td>Organization dependent</td></tr>
<tr><td>Decision Criteria</td><td>Investigate and, potentially, take corrective action when the closure rate or Overdue action items exceed established thresholds <fill in organization specific threshold> or a trend is observed per established guidelines <fill in organizational specific>.</td></tr>
<tr><td>Indicator Interpretation</td><td>• Large deviations for the planned closure may be indicative of larger, more complex tasks ahead or potentially is a sign of challenging personnel interfaces.
• A backlog in the action item closure indicates project risk in terms of rework and/or infeasible schedule, especially if the backlog has higher priority or impact actions.
• If the backlog of action items are related to the technical solution definition, then it indicates there is additional technical risk that should be assessed before proceeding to the next phase, especially if the backlog has higher priority or impact actions. Large number of lingering action items may indicate requirements instability, immature architecture/design, or inadequate stakeholder buy-in. This may be caused by inadequate pre-acquisition systems engineering, including ICD, AoA, AMA (number of review comments; adequate coverage of alternatives in the solution space, etc.)
• The backlog of action items may also be an indication of inadequate quantity or quality (experience or skill mix) of personnel, inadequate project support infrastructure, process maturity/compliance problems, or inadequate project funding.
• Significantly larger number of technical actions assigned at a design review than expected (based on historical data or thresholds) may indicate unacceptable technical risks and may impact readiness.</td></tr>
</table>

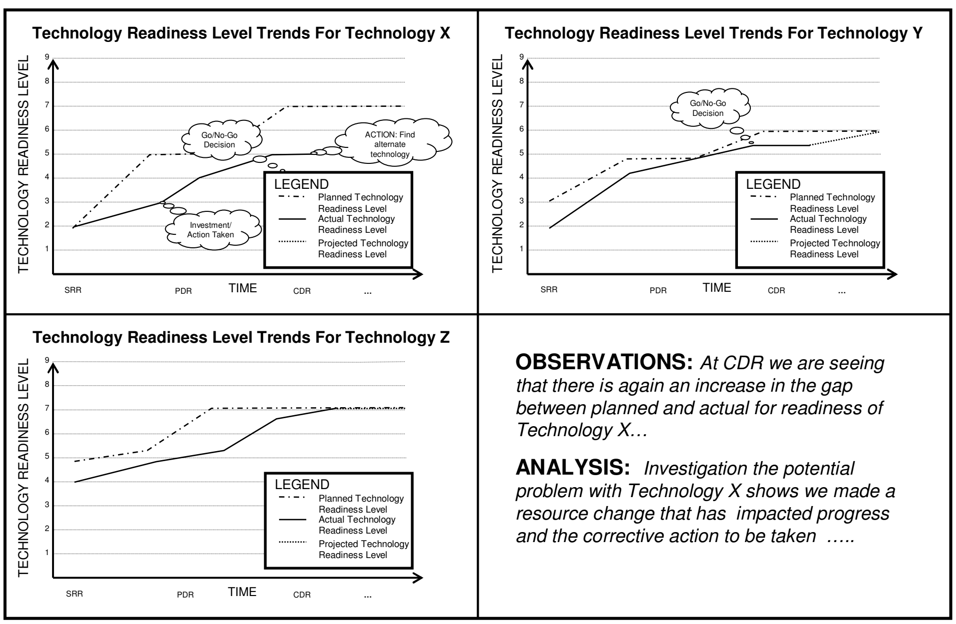
Review Action Closure Trends	
Additional Information	
Related Processes	Review process
Assumptions	• Review minutes/records are maintained & current. • Assumes standard definitions for reviews and life cycle for a project or business.
Additional Analysis Guidance	• Usage is driven by the status of Design Review action item closure. Lower closure than planned, or greater the number of open action items, means higher risk, thus it would be reviewed more frequently. Applies to the Design phase. • Analyze results by the priority of the actions to determine performance on high priority actions that may have the greatest impact. • Analyze the closure rate in conjunction with quality of the action responses (i.e., closure does not equate to quality).
Implementation Considerations	• Includes action items from peer reviews, inspections, technical exchange meetings, in addition to those from large formal reviews/events • Do not sample - collect all Design review action item data. • Should include stakeholder collaboration across life cycle • Ensure common definition of reviews and life cycle • Should use clear, consistent closure criteria for actions
User Of The Data	• Chief Systems Engineer • Product Manager
Data Collection Procedure	• See Appendix F
Data Analysis Procedure	• See Appendix F

3.8 Technology Maturity Trends

This indicator is used to evaluate the trends in technology maturity, including readiness and obsolescence, of specific technologies that are under development. The measure may indicate that technology opportunities exist that need to be examined and may warrant product changes. It may also indicate when a technology is becoming obsolete and may be a candidate for replacement. Trend of obsolescence exposure gives an indication of when to take action due to obsolescence risk. This should help avoid surprises from obsolescence and plan for right timing of technology insertion of new technologies

An example of how such an indicator might be reported is show below for the readiness trends for selected technologies. Refer to the measurement information specification below for the details regarding this indicator; the specification includes the general information which would be tailored by each organization to suit its needs and organizational practices.

Technology Maturity Trends



Technology Readiness Trends. The graph illustrates the actual readiness level of each of three technologies (X, Y, Z) in respect to the planned readiness level. The planned readiness would be determined by factors such as technology investment, availability of component technologies, and other factors. Observations are made on the graphs, with further analysis needed to understand underlying issues and causes where a potential problem is seen. For example, for Technology X, we see that just prior to PDR that there is a significant gap in the actual versus planned readiness, and that additional investment action was taken which post PDR brought the actual readiness much closer to planned, allowing for a go/no-go decision.

3.8.1 Technology Maturity Trend Specification

Technology Maturity Trends	
Information Need Description	
Information Need	Determination of the readiness of new technologies and the obsolescence of currently used technologies in order to maintain a useful and supportable technology base.
Information Category	1. Technology Effectiveness
Measurable Concept and Leading Insight	
Measurable Concept	The potential impact (beneficial or adverse) of technology changes on the future of the project.
Leading Insight Provided	• Indicates that technology opportunities exist that need to be examined and may warrant product changes. A business case needs to be developed to estimate schedules, costs, and benefits (e.g., profit, market share, product performance) of introducing new technology. • Indicates technology is becoming obsolete and may be a candidate for replacement. A business case needs to be developed to estimate schedules (e.g., likely obsolescence dates, time to introduce replacements), costs (e.g., sustaining, development), and benefits (e.g., reduced support costs, improved product performance or customer satisfaction). • Trend of obsolescence exposure gives an indication of when to take action due to obsolescence risk. • Lagging technical maturity progress may provide insight into additional risk of meeting KPPs. • Should help avoid surprises from obsolescence and plan for right timing of technology insertion of new technologies.
Base Measure Specification	
Base Measures	1. Technology Obsolescence Candidates 2. Critical or Beneficial Technology Opportunities 3. Technology Readiness Level (For each new Technology Opportunity) 4. Technology Obsolescence Candidates Realized 5. Technology Opportunity Candidates Realized 6. Expected Time to Realization (Of Technology Readiness or Obsolescence) 7. Actual Time to Realization (Of Technology Readiness or Obsolescence) 8. Expected Cost for Realization (Of Technology Readiness or Obsolescence) 9. Actual Cost for Realization (Of Technology Readiness or Obsolescence) 10. Probability of Technology Insertion or Phase-Out 11. Probable Impact of Technology Insertion or Phase-Out 12. Actual Impact of Technology Insertion or Phase-Out

Technology Maturity Trends	
Measurement Methods	1-3. Count number of Candidates or Opportunities (Based Empirical Analysis and Expert Opinion of the following Sources) A. Industry Contacts and Associations B. Technology Forecast Reports C. Technical Staff 4-5. Count number of Technology Obsolescence Candidates Realized 6. Estimate Time to Realization (Based on Empirical Analysis and Expert Opinion Sources Listed Above) 7. Record Actual Time to Realization (Of Technology Readiness or Obsolescence) 8. Estimate Cost for Realization (Based on Empirical Analysis and Expert Opinion Sources Listed Above) 9. Record Actual Cost for Realization (Of Technology Readiness or Obsolescence) 10. Estimate Probability of Technology Insertion (Based on Empirical Analysis and Expert Opinion Sources Listed Above) 11. Estimate Impact of Technology Insertion (Based on Empirical Analysis and Expert Opinion Sources Listed Above) 12. Record Actual Impact of Technology Insertion (Based on Empirical Analysis and Expert Opinion Sources Listed Above)
Unit of Measurement	1. Technology obsolescence candidates 2. Technology opportunity candidates 3. Technology readiness level 4. Technology obsolescence candidates 5. Technology opportunity candidates 6. Time 7. Time 8. Cost 9. Cost 10. Probability 11. Impact Cost or Schedule 12. Impact Cost or Schedule
Entities and Attributes	
Relevant Entities	• Technology candidates
Attributes	• New Technology opportunities • Existing technology obsolescence • Additional attributes including but not limited to the Process Phases, Disposition Action, Maturity States, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times coupled with the associated events
Derived Measure Specification	
Derived Measure	1. Technology opportunity exposure 2. Technology obsolescence exposure
Measurement Function	1. Technology opportunity exposure: probability * impact (for each opportunity) 2. Technology obsolescence exposure: probability * impact (for each obsolescence candidate)

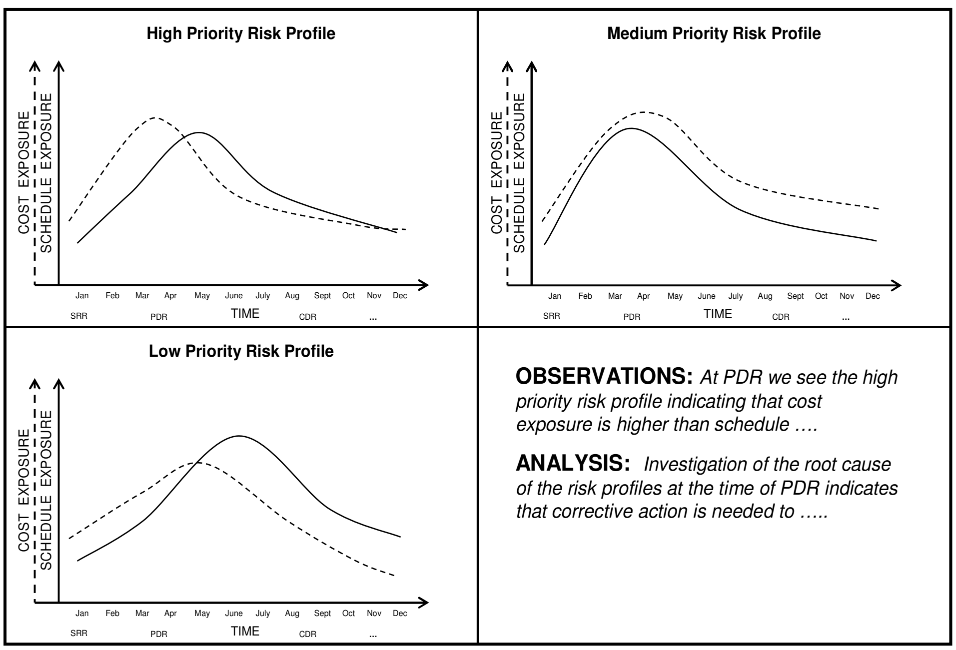
Technology Maturity Trends	
Indicator Specification	
Indicator Description and Sample	• A graph showing trend of technology opportunity exposure, obsolescence exposure and impact of change. • Graph or table showing variances between estimated and actual. • Graph showing trend of technology readiness levels over time.
Thresholds and Outliers	Organization dependent
Decision Criteria	Investigate and, potentially, take action when total technology opportunity exposure, technology obsolescence exposure, and/or an impact of change that exceeds organizational criteria.
Indicator Interpretation	• Provide early warning of potential obsolescence issues • Provide early assessment of impact of changes • Identify when conditions are right to take advantage of new technology opportunities
Additional Information	
Related Processes	Planning, Decision Making, Architectural Design, and Production
Assumptions	Technology opportunities and obsolescence candidates are captured. Technical staff assesses probability, impact, and timeframe of insertion or replacement.
Additional Analysis Guidance	• Collect data for each identified technology opportunity or obsolescence candidate. • Need to consider analysis based on intended life of the system/product.
Implementation Considerations	Use when 1) there is a risk of technology obsolescence that may impact the system; or 2) critical/beneficial technologies are in development. Care should be taken to ensure that "technology push" of introducing "new" technology provides improved value or capability to the customer/consumer/user (unless the business decision and marketing approach is deliberately one of "new technology"). Obsolescence issues may prevent the organization from making/maintaining the product. Need to ask: 1) What can be done with the new technology? – Is the market ready? 2) How can it be incorporated into the architecture and design? 3) What risks are introduced as a result of new technology and product obsolescence? "Best/worst/most likely" cases should be analyzed to understand the spectrum of possible outcomes, their individual likelihood, and the effects on decisions. Reliance on either extreme for technology maturity or obsolescence can lead to suboptimal decisions.
User of Information	• Program/Project Manager • Chief Systems Engineer • Chief Architect • Customer • R&D groups
Data Collection Procedure	• See Appendix F
Data Analysis Procedure	• See Appendix F

3.9 Risk Exposure Trends

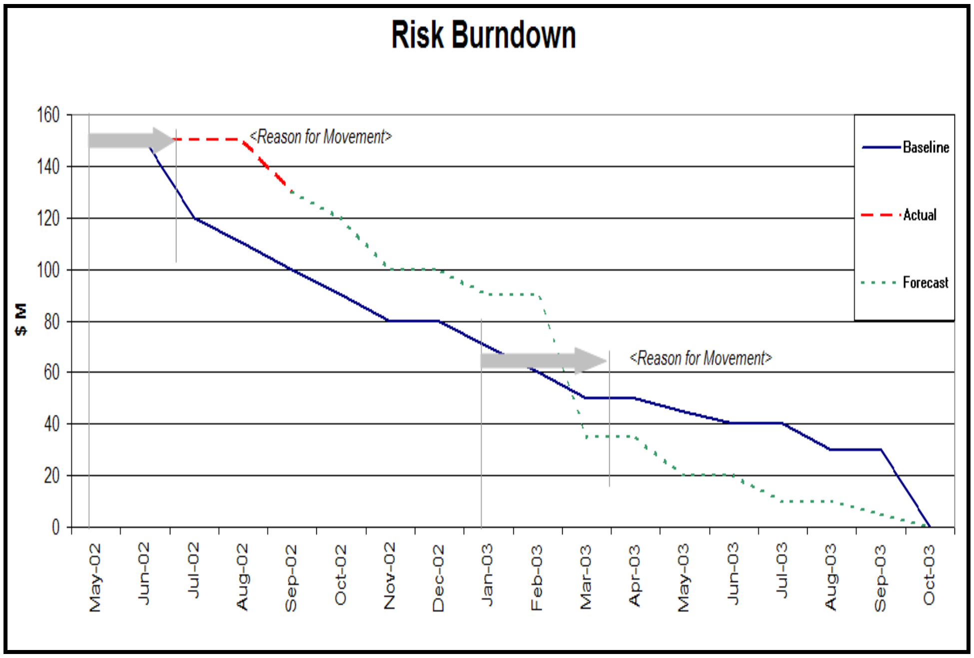
This indicator is used to evaluate the risk exposure over time in terms of cost and schedule, and in context of the level of risk. It indicates whether the project is effectively managing the project risks as shown by predicted exposure ratings over time. If the risk exposure continues to grow or not be reduced, the customer satisfaction will be negatively impacted due to resulting cost, schedule, or technical impacts. It is recommended the Risk Exposure Trends indicators be used in conjunction with the Risk Treatment Trends indicators.

An example of how such an indicator might be reported is show below. Refer to the measurement information specification below for the details regarding this indicator; the specification includes the general information which would be tailored by each organization to suit its needs and organizational practices.

Risk Exposure Trends



Risk Exposure Trends. Given that Cost and Schedule are normalized on the vertical axis graphs above, these graphs illustrates risk profiles of the project in regard to cost and schedule exposure over the life cycle. In this case, profiles for high, medium, and low priority risks are shown separately. The analyst can make certain observations which will require additional analysis to understand what the graphic is showing. For illustrative purposes, cost and schedule exposures are included in this graph. While not included, technical exposure would be another element of this indicator.



Risk Burndown. The graph illustrates the planning and tracking of the risk exposure in terms of cost ($M). The plot of the actual risk exposure burndown shows a slow start. The project team projected the burndown for the remainder of the project to identify whether the risk exposure could be reduced to an acceptable level as the project proceeds and where there were realistic opportunities that could significantly reduce the exposure. To build confidence in the projection, the project team needed to determine the reason for any significant movement (positive or negative). The first movement was due to late project ramp-up and requirements changes. The second movement was where the project team would be able to insert technology to eliminate a set of risks.

3.9.1 Risk Exposure Trend Specification

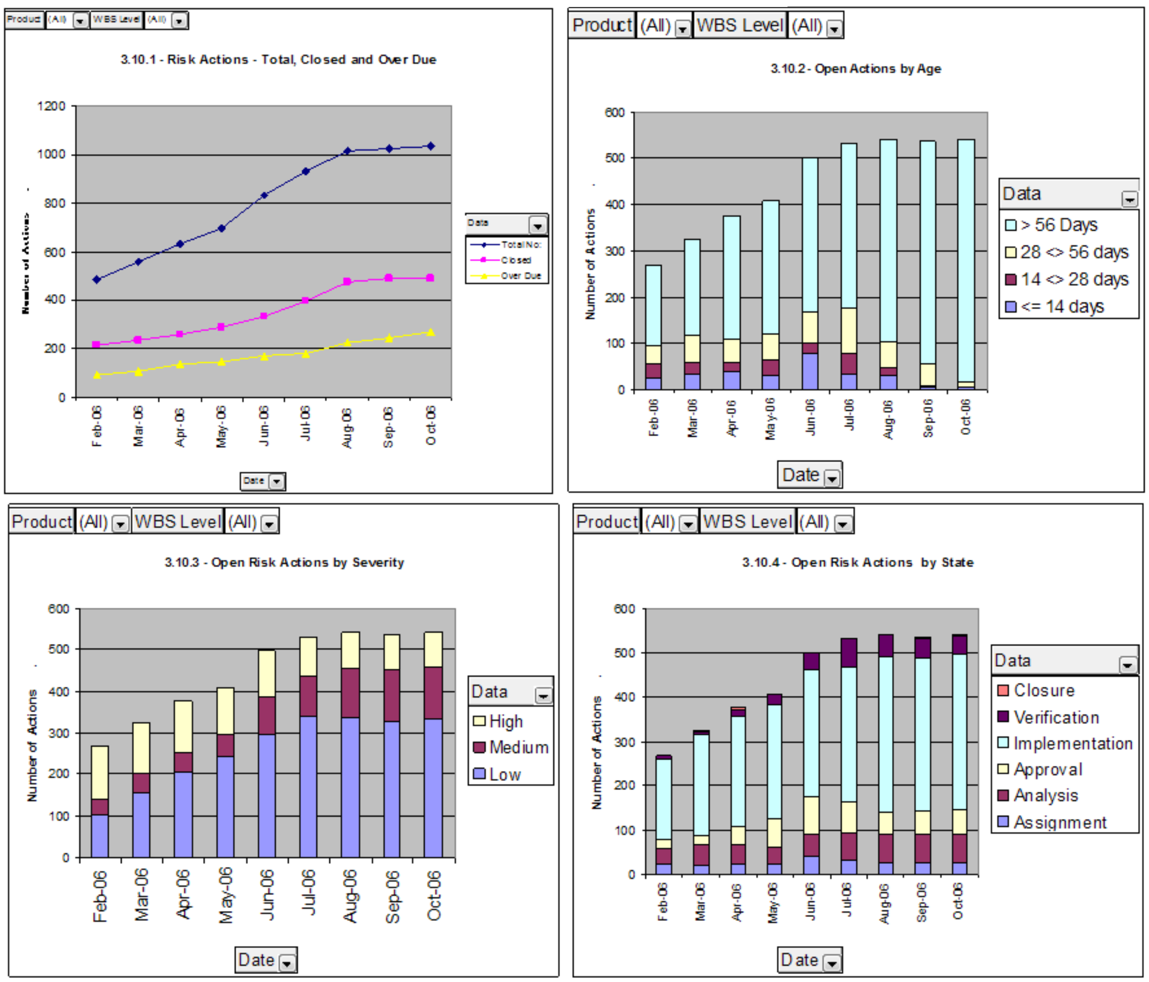
Risk Exposure Trends	
Information Need Description	
Information Need	Determine an estimate of the risk exposure to understand the potential impact to the quality, cost, and schedule of the system solution and the necessary SE effort to manage the exposure.
Information Category	1. Product Quality 2. Schedule and Progress 3. Resources and Cost
Measurable Concept and Leading Insight	
Measurable Concept	Assessment of project effectiveness in managing/mitigating risks • Is the risk exposure going to impact the system solution? • Is the SE effort managing the exposure successfully?
Leading Insight Provided	Indicates whether the project is effectively managing the project risks as shown by predicted exposure ratings over time. • Assessment of risk exposure impacts to the system solution • Assessment of the SE effort in successfully managing the exposure
Base Measure Specification	
Base Measures	1. Risks 2. Probability of Risk Occurrence 3. Impact of Risk Occurrence 4. Criticality of Occurrence (Urgency to Address – If used in a Risk Management process) 5. Planned Actions Per Risk 6. Actual Actions Per Risk
Measurement Methods	1. Count the number of Risks (record associated attributes of interest; e.g., Process Phases, Disposition Action, Maturity States, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times in the risk management repository database) 2-4. Estimate Probability/Risk/Criticality of Occurrence Based on empirical analysis and expert opinion of the following sources A. Industry Contacts and Associations B. Technology Forecast Reports C. Technical Staff Influenced by historical data if any or risk models. 5-6. Record from risk repository database
Unit of Measurement	1. Risks per associated attributes 2. Probability Value 3. Impact Cost or Schedule 4. Rating corresponding to Occurrence Time Interval 5. Handing Actions (of tasks, events) 6. Handing Actions (of tasks, events)
Entities and Attributes	
Relevant Entities	• Risk candidates
Attributes	• Additional attributes including but not limited to the Process Phases, Disposition Action, Maturity States, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times coupled with the associated events

Risk Exposure Trends	
Derived Measure Specification	
Derived Measure	Factored Risk Exposure *[could be in terms of $, time, or technical parameters]*
Measurement Function	1. Probability * Impact *[behavior over time]* 2. Probability * Impact * Criticality *[behavior over time - variant if criticality (or urgency) is used]*
Indicator Specification	
Indicator Description and Sample	1. Risk magnitude/reduction line graph over time that shows trends for each risk category/rating. 2. Table of planned vs. actual risk exposure. • Planned vs. actual over time • Information displayed graphically • See sample charts
Thresholds and Outliers	Organization and/or project dependent.
Decision Criteria	Investigate and, potentially, take corrective action when the exposure trends predict that the risk exposure thresholds are being approached or may become out of control.
Indicator Interpretation	Impact on project execution in meeting Cost, Schedule, Performance, Quality. If the risk exposure continues to grow or not be reduced, the customer satisfaction will be negatively impacted due to resulting cost, schedule, or technical impacts.
Additional Information	
Related Processes	Risk Management, Program Management
Assumptions	• Information is readily available, current, and maintained in a Risk Management repository. • An active risk management effort, which is continuously executed throughout the life of a project, exists.
Additional Analysis Guidance	May use all data or just concentrate on the highest priority risks.
Implementation Considerations	• Align with scheduled reviews (e.g., Risk, IPT, SE, and project) • Aids in identifying trouble spots in terms of performance, cost, and schedule, especially with the collection of categories and sources to share across enterprises to foster lessons learned. Note: For this indicator, the concept of risk does not include opportunities.
User of Information	• Program/Project Manager • Chief Engineer • Risk Manager
Data Collection Procedures	• See Appendix F
Data Analysis Procedures	• See Appendix F

3.10 Risk Treatment Trends

This indicator is used to evaluate effectiveness of handling risks. It indicates whether the project is proactively handling/treating potential problems or risks at the appropriate times in order to minimize or eliminate their occurrence and impacts to the project. If the actions are not closing per plan, then there is a higher probability that risks will be realized. This insight can identify where additional action may be needed to avoid preventable problems or reduce impacts. This indicator may also identify that the project does not have an iterative or continuous process implementation for risk management. Thus, new risks may not be identified and handled, and may affect the project and technical effectiveness/success. Refer to the measurement information specification below for details regarding the indicator. It is recommended the Risk Treatment Trends indicators be used in conjunction with the Risk Exposure Trends indicators.

Risk Treatment



Risk Treatment Trends. As an example of appropriate analysis, consider these four related Risk Treatment trends as a group. Risk Actions, broadly shows that the project is not closing the actions items and also the number of over due actions are increasing. Open Actions by Age, shows risk actions beyond set acceptable thresholds. Open Risk Actions by Severity, might temper any anxiety given the fact that the majority of the actions are of a low and medium severity. Finally, Open Risk Actions by State, gives an understanding that the risk management process is being followed in that the majority of actions are being implemented.

3.10.1 Risk Treatment Trend Specification

Risk Treatment Action Trends	
Information Need Description	
Information Need	Evaluation of risk management project to assess whether the plan/action items have been properly executed.
Information Category	1. Product Quality 2. Schedule and Progress
Measurable Concept and Leading Insight	
Measurable Concept	Assess how successful the SE effort is in mitigating the risks • Are the Risk Treatment/treatment actions being executed and closed as planned? • Is the SE effort driving the closure of the risks?
Leading Insight Provided	Indicates whether the project is proactively handling/treating potential problems or risks at the appropriate times in order to minimize or eliminate their occurrence and impacts to the project. If the actions are not closing per plan, then there is a higher probability that risks will be realized. This insight can identify where additional action may be needed to avoid preventable problems or reduce impacts. This indicator may also identify that the project does not have an iterative or continuous process implementation for risk management. Thus, new risks may not be identified and handled, and may affect the project and technical effectiveness/success.
Base Measure Specification	
Base Measures	1. Risk Treatment Actions
Measurement Methods	1. Count the number of Risk Treatment Actions (record associated attributes of interest; e.g., Process Phases, Disposition Action, Maturity States, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times in the risk management repository database)
Unit of Measurement	1. Action Items
Entities and Attributes	
Relevant Entities	• Risk Treatment Actions
Attributes	• Additional attributes including but not limited to the Process Phases, Disposition Action, Maturity States, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times coupled with the associated events
Derived Measure Specification	
Derived Measure	1. % Risk Treatment Actions closed on time *[per risk level]* 2. % Risk Treatment Actions overdue *[per risk level]* 3. % risks that met risk reduction plan
Measurement Function	1. ((Risk Treatment Actions closed in time interval)/(Risk Treatment Actions planned to close in time interval)) * 100 *[per risk level]* 2. ((Risk Treatment Actions overdue in time interval)/(Risk Treatment Actions planned to close in time interval)) * 100 *[per risk level]* 3. ((Risk reduced in time interval)/(# of risks planned to be reduced in time interval)) * 100

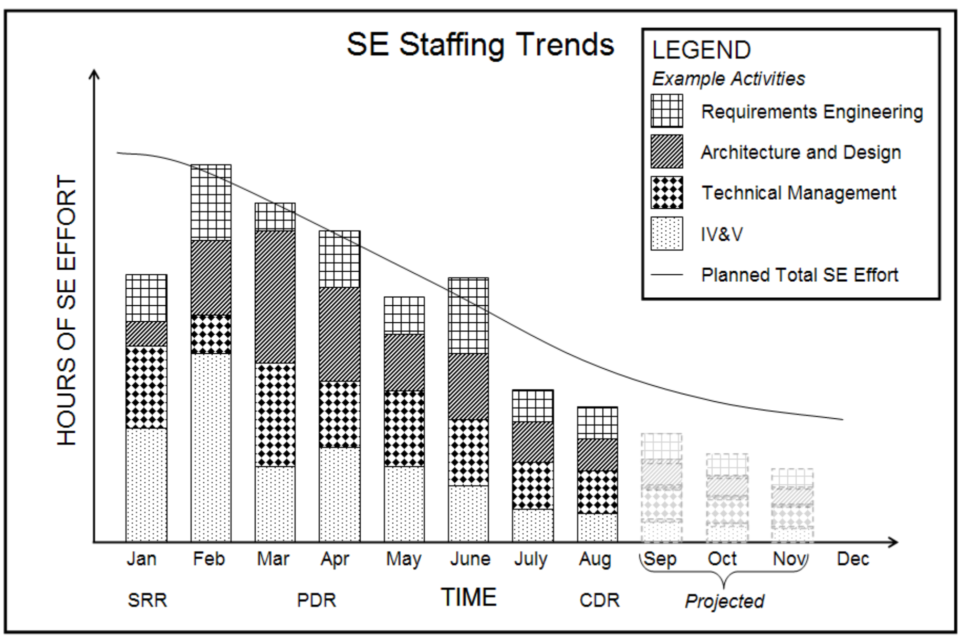
Risk Treatment Action Trends	
Indicator Specification	
Indicator Description and Sample	1. Opened Closed Actions: A line chart plots the number of Actions that are open and closed including Overdue Actions. 2. Age of Actions: A stacked column chart shows the distribution of open actions according to the age of the Risk Treatment action. 3. Actions Priority: A stacked column chart displays the number of open actions that are associated with each of the priority levels. Risk level of associated risks (High, Medium and Low - for filtering purposes to isolate progress on actions for high priority risks. 4. Actions Dispositions: A stacked column chart depicts the number of open Actions that are associated with each of the dispositions (For example, Assigned, Analyzed, Approved, Implemented, Verified and Closed).
Thresholds and Outliers	Organization and/or project dependent.
Decision Criteria	Investigate and, potentially, take corrective action when risk reduction and Risk Treatment action closure are below threshold or expectations. Objective for both is generally near 100%.
Indicator Interpretation	Used to identify whether or not effort is being adequately applied to Risk Treatment/treatment activities. Impact on staffing, planning, development progress, and product delivery. If the actions are not closing per plan, then there is a higher probability that risks will be realized. This insight can identify where additional action may be needed to avoid preventable problems or reduce impacts.
Additional Information	
Related Processes	Risk Management, Program Management
Assumptions	• Information is readily available, current, and maintained in a Risk Management repository. • An active risk management project, which is continuously executed throughout the life of a project, exists.
Additional Analysis Guidance	• May use all data or just concentrate on the highest priority risks. • Effective closure of Risk Treatment actions should positively affect risk exposure.
Implementation Considerations	• Applies to all tasks (i.e., PM, SE, SW, etc.) throughout project life cycle. • Align with scheduled reviews (e.g., Risk, IPT, SE, and project). • The Risk and Opportunity Management process is owned by Program Management and is facilitated for execution by Systems Engineering. Not only are these indicators for Systems Engineering, but they are most likely indicators of overall project performance and health.
User of Information	• Program/Project Manager • Chief Engineer • Risk Manager
Data Collection Procedure	• See Appendix F
Data Analysis Procedure	• See Appendix F

3.11 Systems Engineering Staffing and Skills Trends

This indicator is used to evaluate the staffing and skills mix trends in accordance with plans and expectations. It indicates whether the expected level of SE effort, staffing, and skill mix is being applied throughout the life cycle based on historical norms for successful projects and plans. It may also indicate a gap or shortfall of effort, skills, experience, or turnover that may lead to inadequate or late SE outcomes. The planned staffing can be compared to projected availability through the life cycle to provide an earlier indication of potential risks. It is also a necessary contributor to staff related cost estimates.

An example of how such an indicator might be reported is show below. Refer to the measurement information specification below for the details regarding this indicator; the specification includes the general information which would be tailored by each organization to suit its needs and organizational practices.

Systems Engineering Staff and Skill Trends



Systems Engineering Staffing Trends. The graph illustrates the systems engineering effort versus the planned effort based on historical data and nature of the project. In this graph, the effort is shown in regard to categories of systems engineering activities. We can see that at SRR the data would have shown that the actual effort was well below the planned effort, and that corrective action must have been taken to align actual with planned in the next month of the project.

3.11.1 Systems Engineering Staffing and Skills Trend Specification

Systems Engineering Staffing & Skills Trends	
Information Need Description	
Information Need	Evaluate the adequacy of the SE effort, skills, and experience provided on the project to meet project objectives. Personnel turnover - Projects depend on resources to meet their needs. Significant levels of resource volatility, especially with critical skill types, will cause a ripple effect of inability to meet customer needs, costly overruns, and inability to meet schedule targets
Information Category	1. Resources and Cost – Personnel Effort with respect to Staff Level, Effort, Experience Level, and Staff Turnover
Measurable Concept and Leading Insight	
Measurable Concept	Is SE effort being applied to the project activities consistent with proven organizational or industry practice? Do the staff members have the appropriate skills and experience to achieve assigned tasks? Is the personnel turnover a reason for concern?
Leading Insight Provided	• Indicates whether the expected level of SE effort, staffing, and skill mix is being applied throughout the life cycle based on historical norms for successful projects and plans. • Indicates gap or shortfall of effort, skills, or experience that may lead to inadequate or late SE outcomes. • Planned staffing can be compared to projected availability through the life cycle to provide an earlier indication of potential risks.
Base Measure Specification	
Base Measures	1. Total effort in hours by task, activity, or event (Planned) 2. Total effort in hours by task, activity, or event (Actual) 3. SE effort in hours by task, activity, or event (Planned) 4. SE effort in hours by task, activity, or event (Actual) 5. SE effort in hours by skill and experience (Planned) 6. SE effort in hours by skill and experience (Actual) 7. Number of SE Staff by task, activity, or event (Planned) 8. Number of SE Staff by task, activity, or event (Actual) 9. Number of SE staff leaving the project or company
Measurement Methods	1. Record effort hours from plan by task, activity, or event (may also include experience) 2. Count effort hours by task, activity, or event 3. Record effort hours from plan by task, activity, or event 4. Count effort hours by task, activity, or event 5. Record effort hours from plan by skill and experience (Novice, Junior, Senior, etc.) 6. Count effort hours by skill and experience (Novice, Junior, Senior, etc.) 7. Record the number of SE staff planned for the task, activity, or event 8. Count the number of SE staff actually applied to the task, activity, or event 9. Count the number of SE staff leaving the project or company (by type, skill, or experience)

Systems Engineering Staffing & Skills Trends	
Unit of Measurement	1-6. Hours 7-9. Full-time equivalent staff
Entities and Attributes	
Relevant Entities	• Effort Hours • Skills • Headcount
Attributes	• Task or Activity Type • Experience level (Novice, Junior, Senior, etc.) • Additional attributes including but not limited to the Process Phases, Disposition Action, Maturity States, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times coupled with the associated events
Derived Measure Specification	
Derived Measure	The following may be useful for both the total project and for the specific activities, tasks, or events. 1. % of SE Effort (SE effort / total effort) – Planned 2. % of SE Effort (SE effort / total effort) - Actual 3. % of SE Staffing per plan (SE staffing / total staffing) - Planned 4. % of SE Staffing per plan (SE staffing / total staffing) – Actual 5. Variance of SE Effort (per task and total) 6. Variance of SE Staffing (per task and total) 7. Variance of quantity of SE skills (per given SE skill) 8. Intra-organizational turnover (people leaving to work on another project in the same company) ▪ % of critical systems engineering resources leaving (by type, skill, or experience) 9. Inter-organizational turnover (people leaving to work for another company) ▪ % of critical systems engineering resources leaving (by type, skill, or experience)

Systems Engineering Staffing & Skills Trends	
Measurement Function	1. Planned SE Effort / Planned Total Effort 2. Actual SE Effort / Actual Total Effort 3. Planned SE Headcount / Planned Total Headcount 4. Actual SE Headcount / Actual Total Headcount 5. (Planned SE effort hours) – (Actual SE effort hours) 6. (Planned SE headcount) – (Actual SE headcount) 7. (Planned hours of a given SE skill) – (Actual hours of a given SE skill) [consider experience also, as applicable] 8. Critical systems engineering resources leaving the project (grouped by type, skill, or experience) / Total systems engineering resources leaving the project 9. Critical systems engineering resources leaving the company (grouped by type, skill, or experience) / Total systems engineering resources leaving the company
Indicator Specification	
Indicator Description and Sample	1. Line graphs that show trends of actual SE effort and SE staffing versus plan across the life cycle. Show key events along the time axis of the graphs. 2. Bar charts or stacked bar charts showing the distribution of actual SE effort per task, activity, event or other relevant breakdown against the experiential data for successful projects or against plan. 3. Bar charts showing distribution of actual and planned SE staffing hours by skill. Can use a stacked bar graph to show experience distribution within a skill. 4. Line graphs showing the trends of the most critical SE skills against plan. Show a plan line and actual line over time for each critical skill.
Thresholds and Outliers	Organization dependent
Decision Criteria	Based on the trend, investigate and, potentially take corrective action when the SE effort/skills for a task, event, or portion of the life cycle exceeds established thresholds (positive or negative) or a trend is observed per established guidelines.
Indicator Interpretation	• Lack of meeting planned SE effort or staffing with required skills/experience (i.e., below plan thresholds) potential missed milestones, schedule slips, and/or reduced quality. • Staff hours or headcount that is higher than plan indicates potential cost overrun. • Effort hours, skills and experience should be reviewed together against plan for tasks or activities. This indicates whether the right amount of effort is being applied with the right skills and experience. • Planned staffing can be compared to projected availability through the life cycle to provide an earlier indication of potential risks. • Provides insight into impact of the quantity of systems engineering effort (both hours and headcount) on the overall performance of the project. • Meeting planned effort hours with too few staff will likely result in longer term overtime issues, including impact on cost and quality. • High turnover of SE staff can indicate that there are organizational issues that may adversely change the project.

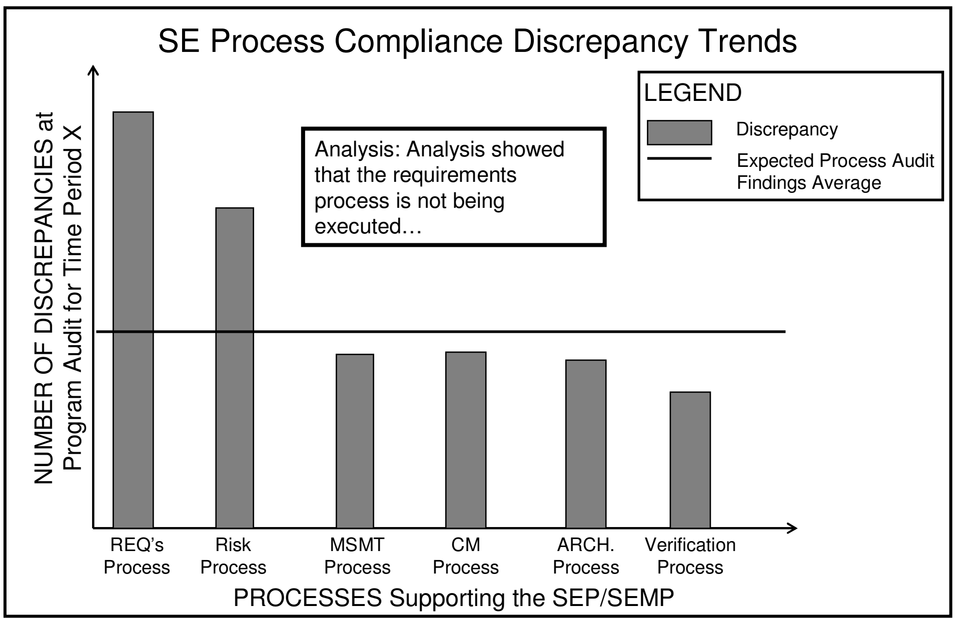
Systems Engineering Staffing & Skills Trends	
Additional Information	
Related Processes	Planning, Control
Assumptions	• Time records that capture SE effort are maintained and current. • SE skill capabilities and experience of personnel are known and maintained. • The Staffing Plan identifies not only roles and quantity, but includes identification of critical skills and when they are needed.
Additional Analysis Guidance	• Can use to aid in trade-off of SE effort versus level/skills. • Should analyze the effort and skills trends together to ensure the right skill mix for the effort.
Implementation Considerations	1. Do not sample - collect all SE effort data and establish applicable distribution. 2. The SE effort is dependent on the tasks/activities the project is responsible for. The project would define the tasks/activities included and would determine whether to track at a total aggregate level or at a lower level. 3. This is most effective, if the distribution of SE skills is identified, an evaluation of personnel against the SE skill set is maintained, and the tracking is performed to ensure the personnel with the right skills are being provided. 4. Consider the utility and importance of staffing measures that span companies through teaming agreements.
User Of The Data	• Program/Project Manager • Chief Systems Engineer • Other Managers
Data Collection Procedures	• See Appendix F
Data Analysis Procedures	• See Appendix F

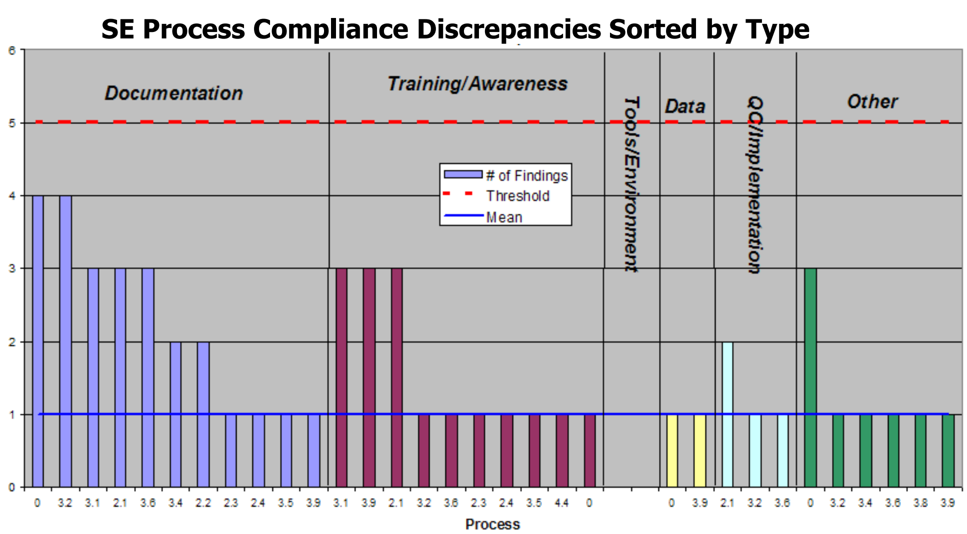
3.12 Process Compliance Trends

This indicator is used to evaluate the trends in process compliance discrepancies to ensure that the project is within expected range for process compliance. It indicates where process performance may impact other processes, disciplines, or outcomes of the project. General non-compliance indicates increased risk in ongoing process performance and potential increases in variance. Non-compliance of individual processes indicates a risk to downstream processes.

An example of how such an indicator might be reported is shown below. Refer to the measurement information specification below for the details regarding this indicator; the specification includes the general information which would be tailored by each organization to suit its needs and organizational practices.

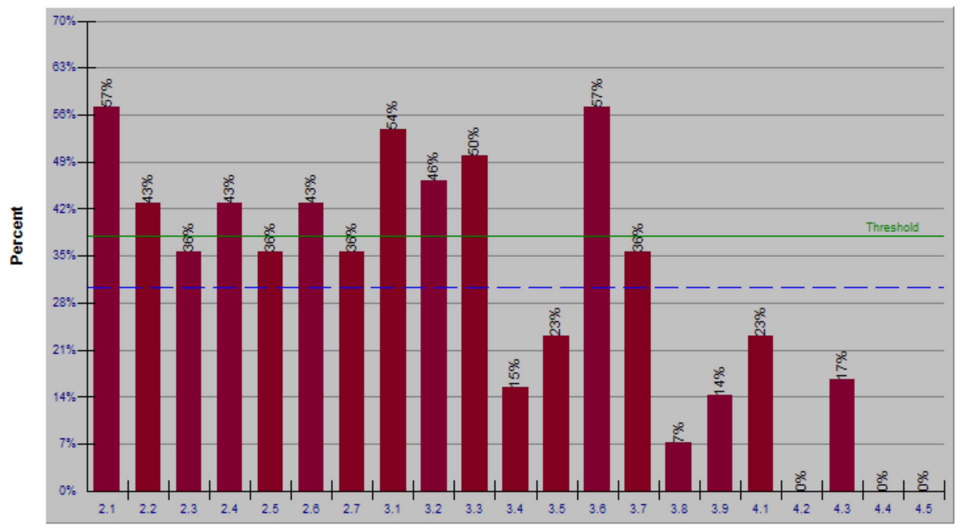
SE Process Compliance Trends





Note: The numbers on the horizontal axis of this figure represent the project processes.

Systems Engineering Process Compliance Trends. The Process Compliance Discrepancy Trends graph illustrates the number of discrepancies for each major process or process area, along with the expected process audit findings based on historical project or organizational audit information. In this case, it can be seen that there are indications that there are issues with the requirements process and the risk process. Further investigation will be needed to determine the root causes – it could be that processes are not being followed, but there could also be cases where there are opportunities for improvement of the process that are needed. As is done in the second figure, it is often useful to sort the discrepancies by type or cause of the discrepancy. In this case, the largest number of discrepancies are caused by issues with the documentation. Issues with the training or lack of awareness of the processes are the next major source of discrepancies in this example. These issues with the documentation or training reflect opportunities for the project or organization to make improvements that will eliminate the risk of errors in future SE performance. The number of discrepancies can give an indication of process performance strength or weakness, helping the project or organization to prioritize improvement efforts.



Process Tailoring. The graph above depicts the amount of tailoring per process at a point in the project. The numbers on the x-axis of the graph are numerical process identifiers. The graph shows a percentage representing the degree of tailoring for each process by the set of projects in the organization. Furthermore, thresholds are set for the acceptable amount of tailoring before needing to investigate whether the needs with respect to the process have shifted. These thresholds might indicate further investigation is needed to determine if there is a systemic problem: a significant project specific process change might indicate the need to update standard process materials or conversely that the specific project will likely have a great deal of difficulty operating within the standard business processes and the accompanying culture. Furthermore, this graphic could be depicted by threshold lines or color-coding. For example, within the acceptable range is depicted in green and exceeding the acceptable range is red.

3.12.1 Process Compliance Trend Specification

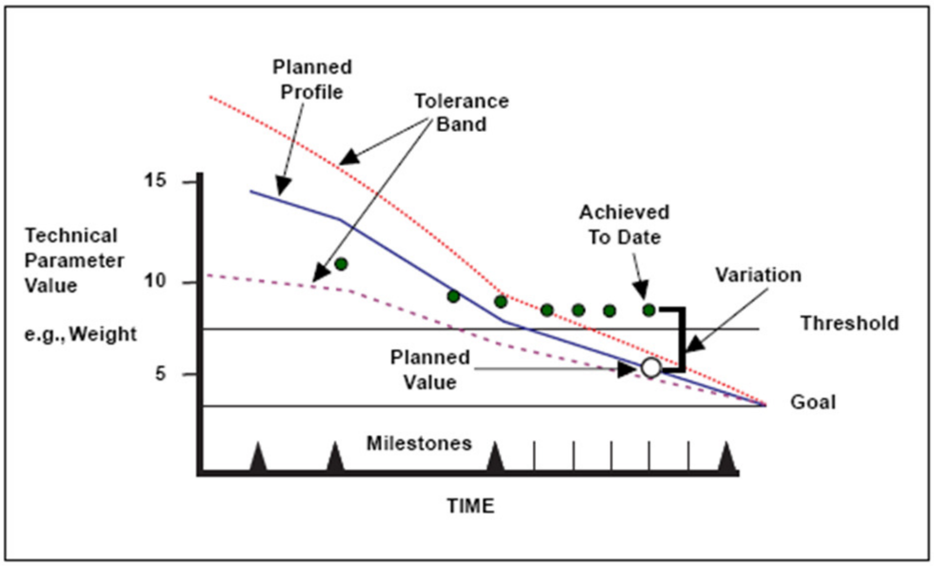
Process Compliance Trends	
Information Need Description	
Information Need	Evaluate project defined SE process performance for compliance and effectiveness.
Information Category	1. Process Performance – Process Compliance and Effectiveness
Measurable Concept and Leading Insight	
Measurable Concept	To what extent are the SE processes in place and being used on the project?
Leading Insight Provided	• Indicates where process performance may impact other processes, disciplines, or outcomes of the project. • General non-compliance indicates increased risk in ongoing process performance and potential increases in variance. • Non-compliance of individual processes indicates a risk to downstream processes.
Base Measure Specification	
Base Measures	1. Tasks or Activities 2. Discrepancies
Measurement Methods	1. Count the number of Tasks or Activities (record associated attributes of interest; e.g., discrepancies, status (satisfied of completed) or tailoring) 2. Count the number of Discrepancies (record associated attributes of interest; e.g., discrepancies severity, status (satisfied of completed) or tailoring)
Unit of Measurement	1. Tasks or Activities 2. Discrepancies
Entities and Attributes	
Relevant Entities	• Tasks
Attributes	• Discrepancy Severity • Discrepancy Classification Type • Additional attributes including but not limited to the Process Phases, Disposition Action, Maturity States, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times coupled with the associated events
Derived Measure Specification	
Derived Measure	1. % Processes with discrepancies 2. Profile of discrepancies 3. High risk processes
Measurement Function	1. Number of processes with discrepancies divided by number of processes (audited) 2. Number of minor, major discrepancies over time 3. Number of processes with major findings or with numerous minor findings

Process Compliance Trends	
Indicator Specification	
Indicator Description and Sample	1. Pareto chart showing quantity of discrepancies for processes from highest to lowest (allows visual identification of those requiring investigation). Show thresholds of expected values based on experiential data. 2. Graph illustrating the number discrepancies or audit findings per process or depicting the amount (percentage or number) of tailoring per process. The data can also be presented to highlight audits findings or process changes categorized according to type, or priority. Furthermore, a business or project might set thresholds for the acceptable amount of findings or tailoring.
Thresholds and Outliers	Organization dependent
Decision Criteria	Investigate and, potentially, take corrective action when the % of processes with discrepancies or number of discrepancies exceeds established thresholds <fill in organization specific threshold> or a trend is observed per established guidelines <fill in organizational specific>. Particularly pay attention to critical processes.
Indicator Interpretation	• General non-compliance indicates increased risk in ongoing process performance and potential increases in variance. • Non-compliance of individual processes indicates a risk to downstream processes.
Additional Information	
Related Processes	All processes
Assumptions	Process audits are conducted and records are maintained & current. Base measures data are available from process audits.
Additional Analysis Guidance	• Usage is driven by the process audit plan • Review together with the work product approval indicators • Although lagging, this indicator also identifies where additional training or quality surveillance may be needed.
Implementation Considerations	• All processes do not need to be audited during all audit periods. Audit those that are most important to success or performed most often during that period. • Need to identify the processes that are downstream from the process observed to provide a leading view. • The lack of a process audit plan is an indicator of risk in this area. • Best to have a non-advocate/independent party involved • Frequency of review is dependent on schedule duration, scope, and composition of the project. • Discrepancy categories are organization dependent • Discrepancy high risk thresholds are organization dependent
User Of The Data	• Chief Systems Engineer • Process Lead • Quality Assurance Manager
Data Collection Procedures	• See Appendix F
Data Analysis Procedures	• See Appendix F

3.13 Technical Measurement Trends

This indicator is used to evaluate the trends in progress toward achieving technical performance requirements. It aids in understanding the risk, progress, and projections regarding a system element or system of interest achieving the critical technical performance requirements.

Refer to the measurement information specification below for the details regarding this indicator; the specification includes the general information which would be tailored by each organization to suit its needs and organizational practices.



Technical Performance Measure. Technical Performance Measure is defined well by the figure above (based on Figure 14.-2 Technical Performance Measurement – The Concept - from Defense Acquisition University's Systems Engineering Fundamentals). Measured values that fall outside established decision criteria (tolerance bands) alert management to take action or perform further investigation. Other relevant terms and relationships are defined as follows:

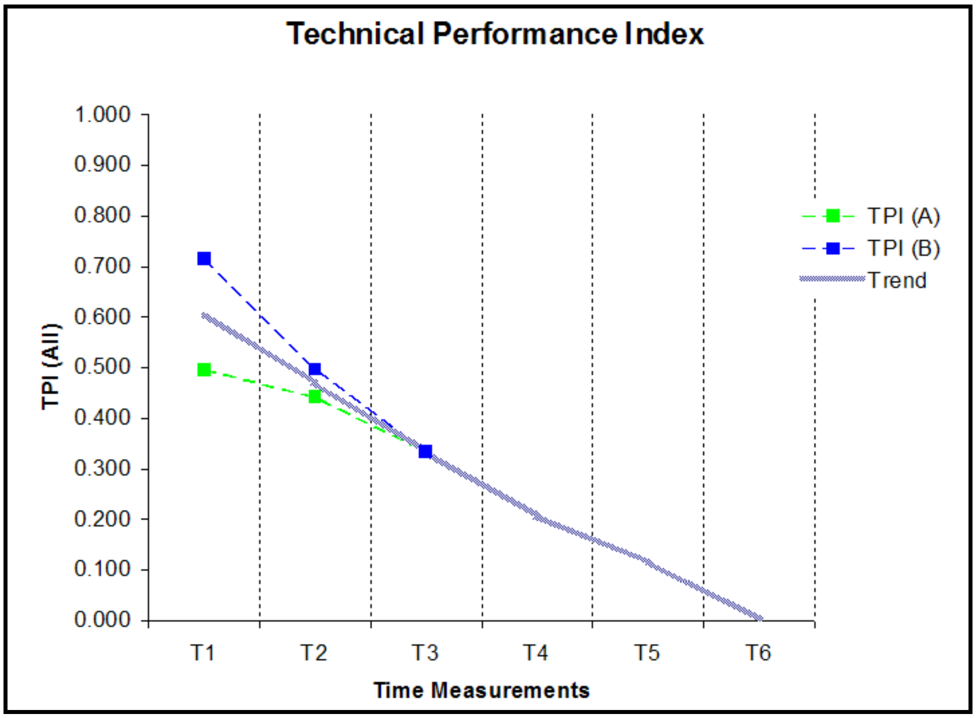
- Achieved-to-Date - Measured technical progress or estimate of progress plotted and compared with the planned progress at designated milestone dates.
- Planned Value - Predicted value of the technical parameter for the time of measurement based on the planned profile.
- Planned Profile - Profile representing the projected time-phased demonstration of a technical parameter requirement. It describes the underlying model of expected behavior of the measures over time.
- Tolerance Band - Management alert limits placed on each side of the planned profile to indicate the envelope or degree of variation allowed. The criteria are used to trigger action or further investigation. Tolerance bands are an acknowledgement of estimating error and reflect acceptable risk limits associated with achieving the performance measured by the TPM.
    - Threshold - The limiting acceptable value of a technical parameter; usually a contractual performance requirement.
- Demonstrated Technical Variance – the difference between the 'Planned Value' and the 'Achieved-to-Date' (or demonstrated/measured) value at a specific point in time.
- Predicted Technical Variance – the difference between the 'Planned Value' at EOC and the 'Current Estimate' of the parameter.

Technical Performance Measures

Technical Performance Measures			Current							
	TPM 1	G	G	G	Y	Y	Y	G	G	G
	TPM 2	G	G	G	G	G	G	G	G	G
	TPM 3	G	G	R	R	R	R	Y	Y	Y
	TPM 4	G	G	G	G	G	G	G	G	G
	TPM 5	G	Y	Y	Y	Y	Y	G	G	G
	TPM 6	G	Y	Y	Y	G	G	G	G	G
	TPM 7	G	Y	Y	G	G	G	G	G	G
	TPM 8	G	Y	Y	G	G	G	G	G	G
	TPM 9	G	Y	R	R	R	R	R	Y	Y

Time

The technical performance measures table depicts a projects key or critical TPMs, and the status of these measures over time. The trend is the number of key or critical TPMs in each status color Red, Yellow, or Green as declared by the project. The important requirement is that the criterion for the status of the TPMs is standardized by the business.



Technical Performance Index. One of the contributing businesses has developed a Technical Performance Index (TPI). The index is based on the business's own defined mathematics and logic to calculate an "aggregate" trend quantifying and forecasting an overall system's performance. It provides a method to visualize aggregate system performance achievement in one graphic. For each TPI, the deviations of all the contributing TPMs are normalized from the associated thresholds.

The index has successfully enabled discussions of programmatic technical issues, by appropriately simplifying the project details for non-technical settings while still retaining the ability to drill-down to lower tiered levels to understand problem areas with trend data. Furthermore, the TPIs depict the project trend for achieving overall technical performance parameters and the extent of performance lag. This aids in the identification of the risk-driving TPMs and in the project prioritization of focus to improve technical performance.

3.13.1 Technical Measurement Trend Specification

Technical Measurement Trends	
Information Need Description	
Information Need	Understand the risk, progress, and projections regarding a system element or system of interest achieving the critical technical performance requirements.
Information Category	1. Technology Effectiveness - Technology Suitability and Volatility 2. Product Quality
Measurable Concept and Leading Insight	
Measurable Concept	To what extent are the performance parameters feasible and being achieved per plan?
Leading Insight Provided	Indicates whether the product performance is likely to meet the needs of the user based on trends. • Project the probable performance of a selected technical parameter over a period of time • Project the probable achievement of system balance (satisfaction of all TPMs). Indicates feasibility of alternatives and impact of potential technical decisions. • Assessments of the project impact for proposed change alternatives Provides insight into whether the system definition and implementation are acceptably progressing. • Early detection or prediction of problems requiring management attention • Allows early action to be taken to address potential performance shortfalls (transition from risk management to issue management).
Base Measure Specification	
Base Measures	Specific base measures are dependent on the MOE/MOP/TPM; general base measures are: 1. Values of Technical Measure
Measurement Methods	1. Record Estimated Values of the MOE/MOP/TPM (record associated attributes of interest; e.g., Maturity States, Process Phases, Maturity States, Priority Levels) Based on empirical analysis and expert opinion of the following sources A. Industry Contacts and Associations B. Technology Forecast Reports C. Technical Staff 2. Record Actual Values of the MOE/MOP/TPM (record associated attributes of interest; e.g., Maturity States, Process Phases, Maturity States, Priority Levels)
Unit of Measurement	Depends on MOE/MOP/TPM - measured values (e.g., miles, pounds, watts, MTBF, etc.)
Entities and Attributes	
Relevant Entities	• Technical Requirements
Attributes	• Attributes are dependent on the MOEs/MOPs/TPMs chosen • Additional attributes including but not limited to the Process Phases, Disposition Action, Maturity States, Priority Levels, Cause, Impact Level, Classification Type, and Dates & Times coupled with the associated events

Technical Measurement Trends	
Derived Measure Specification	
Derived Measure	1. Delta performance (planned vs actual) 2. Delta performance to meeting thresholds and objectives 3. % Sum|normalized deviations from plan| across all measures
Measurement Function	1. Delta $performance_{plan}$ = Planned performance – Actual performance 2. Delta $performance_{threshold}$ = Threshold performance – Actual performance 3. Sum |Delta $performance_{plan}$/Planned Performance|
Indicator Specification	
Indicator Description and Sample	Trends graphs/charts of MOEs (or KPPs), MOPs, TPMs, and margins. Graphical representation will be dependent on the specific MOE/MOP/TPM chosen.
Thresholds and Outliers	Organization and/or contract dependent.
Decision Criteria	Investigate and, potentially, take corrective action when the values of the MOEs/MOPs/TPMs exceed the tolerance bands (e.g., acceptable risk range) <fill in MOE/MOP/TPM specific tolerance band values> or a trend is observed per established guidelines <fill in specific details>.
Indicator Interpretation	• Technical progress behind plan indicates that risk is increasing. Technical progress that violates the defined "tolerance band" creates an issue to be managed with corrective action. • Technical progress ahead of plan indicates risk is decreasing. Technical progress that satisfies the objective effectively closes the risk.
Additional Information	
Related Processes	Technical Risk, Requirements Analysis, Modeling, Design and Integration
Assumptions	MOE/MOP/TPM measurement records are maintained & current. This includes accurate and current measured values from analysis, prototype, and test.
Additional Analysis Guidance	See Technical Measurement Guide (PSM, INCOSE)
Implementation Considerations	• TPMs should be derived from KPPs or other critical requirements that affect the technical success of the project. • Action strategy for failure to remain within defined profiles should be defined ahead of time (risk mitigation planning) to improve likelihood of implementation and avoid management paralysis. Mitigation plans should consider any coupling to other TPMs. • Comparisons of achieved results vs. needed profiles must be based on the same criteria, scenario, etc., to avoid "gaming". • TPMs should be reported with error tolerances to indicate the confidence level or uncertainty of the analysis models or test results. • It is useful to understand the MOE/MOP/TPM sensitivity to changes in other parameters. • Solid Systems Engineering Foundation - Staff, Requirements Analysis, Architecture, Implementation, Integration, Verification, Facilities.
User Of The Data	• Chief Systems • Chief Systems Engineer • Product Manager • Quality Assurance Manager

Technical Measurement Trends	
Data Collection Procedures	• See Appendix F
Data Analysis Procedures	• See Appendix F

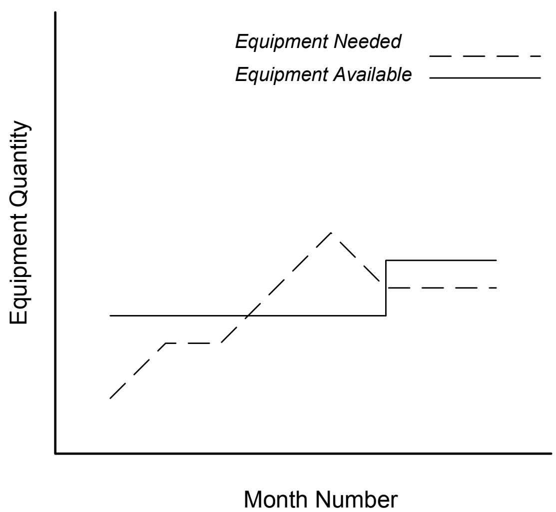
3.14 Facility and Equipment Availability

This indicator is used to determine the availability of critical facilities and equipment needed for systems engineering activities over a project's lifecycle. The indicator is composed of two metrics, one type that measures facility availability and another that measures equipment availability. The indicator provides a view of facility and equipment availability over time.

Facilities of various types may be required. Each facility type may provide key capabilities. The facility availability metric would measure the difference between the planned need for a facility type and the existing inventory of available facilities that meets the need for the desired capability.

Equipment of various types may be required. Each equipment type may provide key capabilities. The equipment availability metric would measure the difference between the planned need for an equipment type and the existing inventory of available inventory that meets the need for the desired capability.

The figure below shows how such an indicator may be reported for equipment availability. Facility availability can be shown in a similar manner.



3.14.1 Facility and Equipment Availability Trend Specification

Facility and Equipment Availability	
Information Need Description	
Information Need	**Facility Availability** - Identifies where there may be issues with the projected availability of critical facilities for a project. Insufficient facilities of various types may cause a project to be unable to meet its customer needs, create costly overruns, and inability to meet schedule targets. Facilities may include labs, test ranges, floor space, etc. **Equipment Availability** - Identifies where there may be issues with the projected availability of critical equipment for a project. Insufficient equipment of various types may cause a project to be unable to meet its customer needs, create costly overruns, and inability to meet schedule targets. Equipment may include software and systems applications used throughout the project lifecycle. Examples include fabrication equipment (i.e. for prototypes), measurement equipment, cleanroom equipment, test equipment, etc.
Information Category	• Resources and Cost
Measurable Concept and Leading Insight	
Measurable Concept	Assesses whether adequate facilities and equipment can be allocated to the project to meet the lifecycle milestones Facility availability: • Difference between systems engineering need on the project and available facilities based on based on projected needs Equipment availability: • Difference between systems engineering need on the project and available equipment based on projected needs
Leading Insight Provided	• Indicates potential shortfalls of systems engineering related facilities and equipment • Indicates potential problems with the project's ability to meet desired milestones
Base Measure Specification	
Base Measures	Facility availability (by type/capability) • Facilities required • Facilities available • Facilities differences Equipment availability (by type/capability) • Equipment required • Equipment available • Equipment differences

Facility and Equipment Availability	

Measurement Methods	• Previous history from similar projects to help determine projection trends
	
	• <u>Facility availability</u>
	  o Forecast (count) the facilities required (performed during project planning)
	  o Determine the facilities available over the schedule need
	  o Assess the difference between the projection and availability
	
	• <u>Equipment availability</u>
	  o Forecast (count) the equipment required (performed during project planning)
	  o Determine the equipment available over the schedule need
	  o Assess the difference between the projection and availability

Unit of Measurement	• <u>Facility availability (by type)</u>
	  o Facility quantity/floor space
	  o Facility quantity/floor space
	  o Facility quantity/floor space
	
	• <u>Equipment availability (by type)</u>
	  o Equipment quantity
	  o Equipment quantity
	  o Equipment quantity

Entities and Attributes	
Relevant Entities	• Facility • Facility user (lease, borrow, purchase) • Facility manager/controller/seller/agent • Equipment • Equipment user • Equipment manager/controller/seller/agent
Attributes	• Facility type needed • Facility type inventory • Equipment type needed • Equipment type inventory
Derived Measure Specification	
Derived Measure	• Profile showing alignment of specific project needs to company-level needs for facility type • Profile showing alignment of specific project needs to company-level needs for equipment type
Measurement Function	• Facility type needed – facility type available • Equipment type needed –equipment type available
Indicator Specification	
Indicator Description and Sample	• Profile indicating facility type needed vs. availability over time • Profile indicating equipment type needed vs. availability over time

Facility and Equipment Availability	
Thresholds and Outliers	• Project and organization dependent
Decision Criteria	• Evaluate and reconcile differences
Indicator Interpretation	• Used to understand facility and equipment availability
Additional Information	
Related Processes	• Staffing and skills trends required to operate and use facilities and equipment • Planning, assessment, and control
Assumptions	• Facility and equipment inventory is maintained. • Multiple project needs for facilities and equipment are tracked for an organization. • Coordinator(s) exists to monitor and allocate facilities and equipment.
Additional Analysis Guidance	• Assessment of needs should track to the project's schedule. This implies that schedule slips will affects the timing of when resources are required. • Contention for resources will have a negative effect on project schedule. • Higher conflicts between needs and available resources across projects mean higher risks. • Prioritization process is important at the organizational level to de-conflict facility and equipment requirements across projects. • Lifecycle phase is critical in determining prioritization and segmentation of facilities and equipment.
Implementation Considerations	• Must have a firm grasp on the inventory of facilities and equipment as well as a solid process for collecting the project needs.
User of Information	• Systems engineering manager • Chief systems engineer • Facility / Site of Development and Resource Deployment • Project Manager • Company executives
Data Collection Procedure	See Appendix F
Data Analysis Procedure	See Appendix F

3.15 Defect and Error Trends

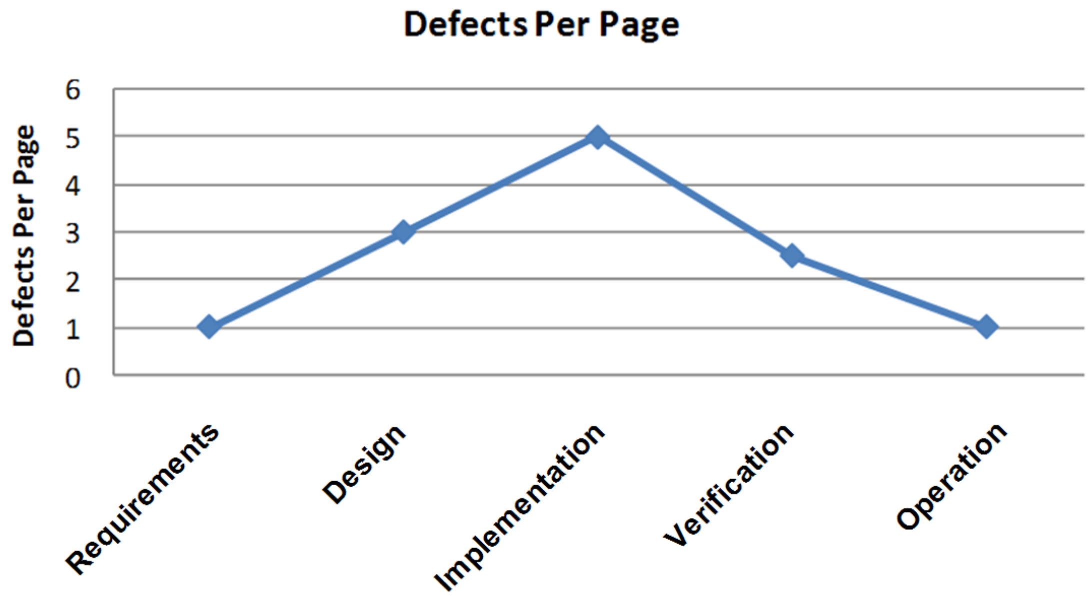
This indicator is used to evaluate the trends associated with defects and errors. It also describes the use of escape or leakage trends as leading indicators. This indicator provides insight into the rate of errors typically found in different system definition artifacts based on the historical experiences.

> A "defect" is a deviation of a product at any stage of its development, implementation, or operation from its requirements or applicable standards. A "defect" can occur in the creation of almost any physical or intellectual product or service such as a report, paper, mechanical drawing, physical construct, or software. The term "defect" can also be applied to the performance of a service such as cleaning the floor in a building. The terms "escape or "leakage" refer to defects that are not identified during the development stage in which they are created.

This technique has used successfully in the case of software development but can be applied to the creation of many engineering artifacts. This introduction describes the general approach and applies it as an example to the creation of a generic document, such as a requirements document. It could easily be applied to the creation to the architecture diagrams, SysML models, or analysis/trade-study reports given appropriate analysis of historical data. Following the general example, a software implementation is discussed. We have used this software focused example because the use of defect management, including defect discovery planning, tracking, and estimation is much more mature for software engineering than for other products or services.

Using a generic development and lifetime process of: requirements definition and analysis; design; implementation; integration; verification; transition; validation; operation; maintenance; and disposal, we will now look at the creation of a Systems Engineering document. During the requirements stages, the customer and the document creator define the purpose of the document, whom it is to address, and other matters. There could be various types of defects created at these stages that may or may not be recognized before the document is completed. For example, a requirement might be omitted. Such a defect might not be found before the stage is completed and all of the stakeholders have agreed that it is acceptable. If this is the case, the defect might "escape" and might not be "detected" until the document is verified or reviewed after it has been completed. A worse outcome is one in which the defect is not found until the operation stage, that is, until after the document has been published and distributed.

Continuing with the document creation scenario, during the design stage, the outline of the document is created and some detail is created. Sometimes, such outlines or designs are peer reviewed and defects are discovered before they escape to later stages. During the implementation stage, the document is actually written and is scrutinized by its authors who may find various defects in the form of improper English, spelling errors, and so on. During the testing of the verification stage, persons other than the authors very carefully review the document, looking for defects in it, such as omissions of sections that had been agreed upon.



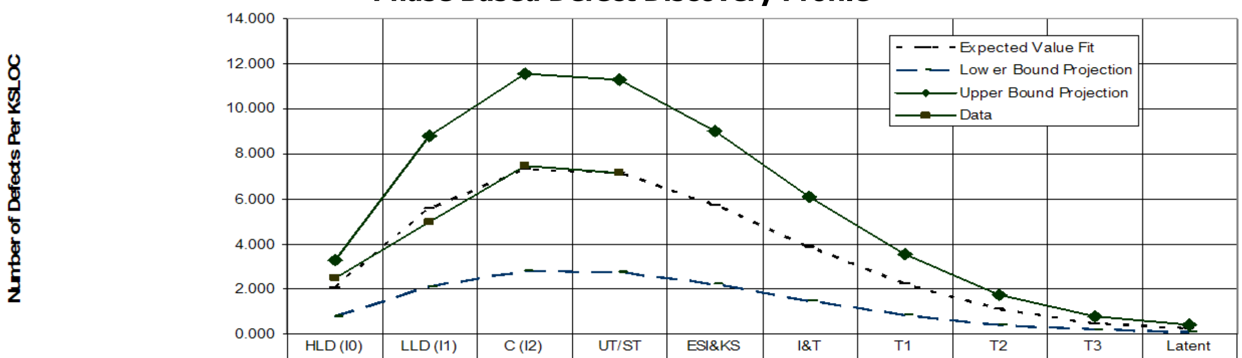
Defects Per Page. For one instance of document creation, one could plot the defects per page as shown in the graph above. Based on the defect discovery experience of a large number of instances, one could plot the upper and lower bounds and the expected values. Also, a defect projection mathematical model might be used to estimate likely defect discovery in later stages of the document development. Then, as the development of a new document is started, its defect discovery experience could be plotted in contrast. Deviations from what is expected could be leading indicators of a problem. Recognizing such a problem could cause various mitigating actions to be taken, such as performing a very thorough review of the document at the stage at which the unexpected behavior was found.

A software defect is a deviation of the software representation at any stage of development or operation from its requirements and/or applicable standards. Thus, there can be requirements defects, design defects, and code defects.

For example, as depicted above, a defect might have been created in detailed software design, but not discovered during design inspection or peer review. Rather, it might have escaped to be implemented in code, and in a worst-case scenario was not discovered until the code had been placed in operation; potentially impacting the operation of the system.

There are two principal types of software defect discovery-prediction models: activity or phase-based and time-based. Both types of models use counts of defects that occur over some period of time or during some activities or phases of the software development and test process to estimate or predict measures such as the number of defects that can be expected during a subsequent period of time (including post-delivery) or subsequent phase (including post-delivery or latent). A phase-based model can be used to obtain estimates of defect discovery for the testing phases and for the post-delivery or latent "phase" commencing during the early stages of software development. These early stage discovery figures are thus good leading indicators of defect discovery in later stages of software development if corrective action is not to remove some of the defects in the software and thus get the software defect discovery profile back on track.

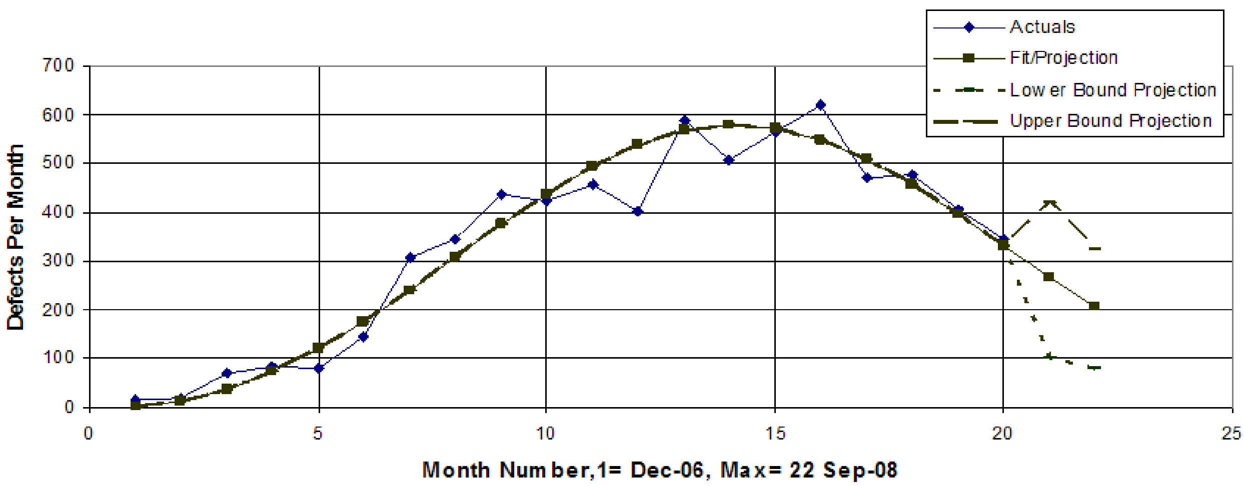
Phase Based Defect Discovery Profile



	HLD (I0)	LLD (I1)	C (I2)	UT/ST	ESI&KS	I&T	T1	T2	T3	Latent
Expected Value Fit	2.094	5.578	7.328	7.180	5.736	3.868	2.239	1.123	0.491	0.277
Low er Bound Projection	0.805	2.144	2.816	2.759	2.204	1.486	0.860	0.432	0.189	0.106
Upper Bound Projection	3.300	8.789	11.547	11.314	9.038	6.094	3.527	1.770	0.774	0.436
Data	2.50	5.00	7.50	7.18						

Phase

Time Based Defect Discovery Profile



Deflect Discovery Profiles. The first figure is an example of phase-based defect discovery profiles, showing phase-by-phase output plots. The second figure is an example of time-based model defect discovery profiles, showing time interval output plots. The phase-based defect discovery profiles can cover the entire development life-cycle. The time-based defect discovery profiles can cover the portion of the life-cycle, for example during the integration test phase.

3.15.1 Defect or Error Trend Specification

Defect/Error Trend	
Information Need Description	
Information Need	1. Understand the proportion of defects being found at each stage of the development process of a product or the execution of a service 2. Understand opportunities for finding defects earlier in the development process and reducing the number of defects created
Information Category	1. Product Quality – Functional Correctness 2. Process Performance – Process Effectiveness 3. Product Size and Stability – Functional Size and Stability
Measurable Concept and Leading Insight	
Measurable Concept	• Is the SE effort proceeding towards the creation of a product or the delivery of a service that meets the quality expectations of its recipient? • Is the phase containment of defects adequate? • Is the defect discovery adequate?
Leading Insight Provided	• Indicates whether emerging product will meet quality objectives • Indicates whether a change in defect discovery (verification) process might be of value
Base Measure Specification	
Base Measures	1. Number of defects found at each discovery (verification) stage 2. Amount of product in which the stated number of defects were revealed in each discovery (verification) stage
Measurement Methods	1. Record number of defects discovered at each defect discovery (verification) stage 2. Record amount of product in which defects discovered at each discovery (verification) stage
Unit of Measurement	1. Count of defects found 2. Size of unit in which defects are found (e.g., number of pages, number of source lines of code; size of drawing; number of scenarios)
Entities and Attributes	
Relevant Entities	• Product defect forecast
Attributes	• Changes in quality technical performance
Derived Measure Specification	
Derived Measure	1. Estimated number of latent defects (number of defects delivered in product) 2. Total number of defects created during the development process 3. Location of peak of the defect discovery profile, its phase index number, where phase 1 goes from phase index numbers 0 to 1, phase 2 goes from phase index number 1 to 2, etc.
Measurement Function	1. Weibull model functions can be used to fit defect discovery data, such as shown in Figures 2 and 3. A member of the Weibull "family" that is often found to be useful is the Rayleigh model. The cumulative form of this model is: N(t)=E*(1-(exp(-b*(t^2))), where: b=0.5/(tp^2), t=x-axis value, and tp=location of the peak.

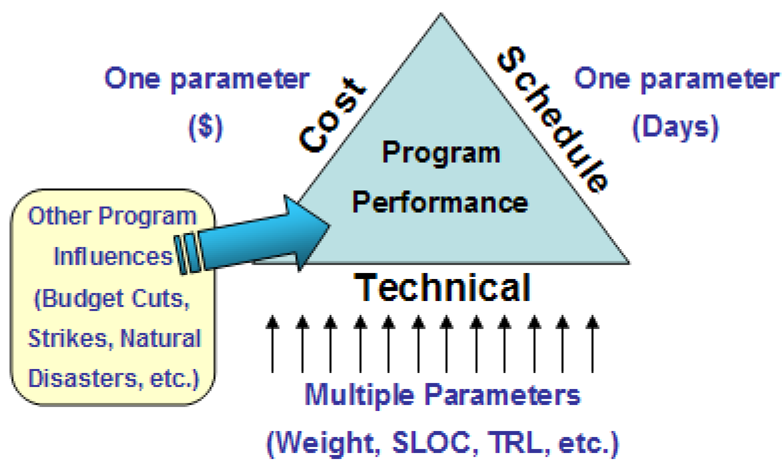
Defect/Error Trend	
Indicator Specification	
Indicator Description and Sample	The defect discovery profile includes a fit to defect data as it becomes available and projections to later phases or later time intervals based on the data fits for earlier phases or earlier time intervals; thus, providing a leading indicator of expected defect later (phase or time interval) defect discovery. This information can be used to determine whether defect discovery will meet goals or not. If not, various mitigating actions can be taken, such as re-inspecting code, should the defect discovery during code inspections not have met expectations.
Thresholds and Outliers	Range of acceptable values for Defect discovery or defect discovery per unit size at each stage, based on past project history (e.g., a process performance model)
Decision Criteria	Does the number of defects found at a particular phase about what was expected for each phase itself as well as with respect to adjacent phase, e.g., value 1>value2<value3 rather than value1<value2<value3.
Indicator Interpretation	• The defect discovery profile (fit to available data plus projections to later phases or time intervals) can be less than, equal to, or greater than expected (within some previously established) tolerance. Values exceeding tolerance are calls to action which can include, re-inspections, providing additional training to the personnel performing the verification steps, etc.
Additional Information	
Related Processes	Reliability estimates - the values of estimated latent or delivered software defects are key inputs into reliability and availability estimates.
Assumptions	• The verification steps are performed consistently well such that the fits/projections of defect data are valid.
Additional Analysis Guidance	• Defect models used should used as part of a closed-loop defect management process in which: 1. Defect discovery goals are planned, consistent with customer requirements and process capability (as indicated by past projects' defect discovery profiles); 2. Actuals and projections are tracked against goals; 3. Corrective/mitigating action is taken if the goals are not met (within stated tolerances).
Implementation Considerations	• Assure the use of properly trained personnel to discover defects, e.g., inspector, testers
User of Information	• Development Personnel • Chief Systems Engineer • Quality Assurance Personnel • Program/Project Manager • Customers
Data Collection Procedure	• See Appendix F
Data Analysis Procedure	• See Appendix F

3.16 System Affordability Trends

This indicator is used to estimate the affordability of a system. The system affordability is defined by an estimate of the cost of the system at the end of the project with a given confidence and the customer's ability or willingness to pay that price for the project's deliverables. Each specific estimate is made as if the associated baseline assumptions are maintained until the end of the project. It thus provides an advanced view of what is expected to be reality at the end of the project. However, every project is influenced by external forces and internal changes of assumptions; accordingly the estimate of system affordability will change over time.

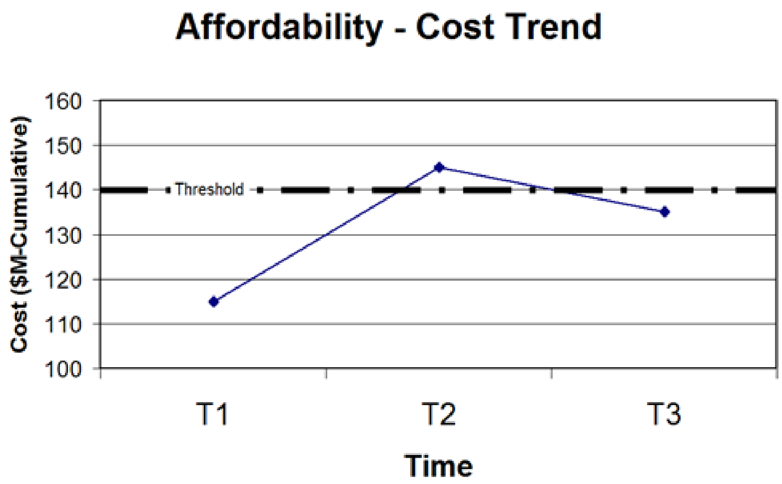
Furthermore, this confidence is the probability that the system's cost will be less than or equal to some stated baseline value. It provides insight as to whether the given set of needs can be met within the stated budget. Accordingly, the indicator illustrates the relationship and sensitivity between the source of uncertainty and the associated impact. It can also be used to characterize the ability of an organization to deliver a system at a level of affordability that should allow a customer to acquire the system and accordingly the indicator provides insight to the ultimate economic viability of a project.

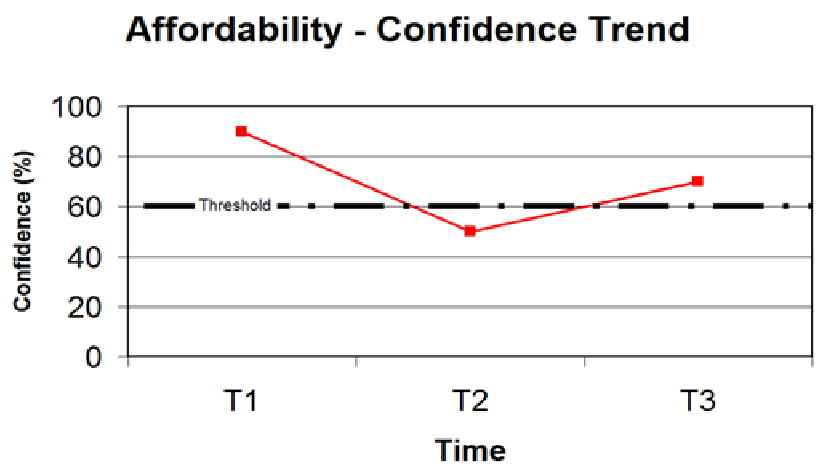
Finally, the system affordability should be determined at several times during the course of a project. Accordingly, with each successive system affordability estimate, it should be compared and contrasted to expected values established from the baseline set of budgets. The figure below depicts a relationship between project performance, in this case measured in terms of Affordability, as a balance of technical, cost, and schedule concerns. It also represents the reality that external factors also have an influence the project performance.



Affordability, as used here, refers to the probability or confidence of achieving a stated set of needs at a stated cost and schedule. The treatment of risk (*risk defined as 100%-confidence*), risk associated with cost, schedule, or individual characteristics of performance is determined (often estimated) on the basis of an organization's ability to provide that characteristic and should take in to account the project's phase (e.g. if its being estimated at pre-concept, concept, early design, design or implementation). Accordingly, affordability is a composite of several other indicators and measures. Typically, the combination of these indicators and measures may carry different levels of importance. When they are combined mathematically, different weights can be assigned to them, corresponding to their relative importance. For example, in the extreme, a project may be principally cost or schedule driven.

It is worthwhile noting that usually customers have constrained budgets and can only tolerate certain levels of risk; although they may not be able to explicitly state what "risk" they can tolerate or "confidence" they desire. For that reason, cost dominates the discussion with the customer. Consequently it is common to normalize the expression of affordability in terms of cost. Typically, schedule and performance as well as the associated effort drive cost.





Affordability Trends. In the example depicted above, we see a sequence of the affordability distributions determined at a succession of times during the course of the project. The initial affordability distribution is set at that the baseline cost and confidence for the stated requirements. If based on organizational capability and other factors, the supplier has estimated that the system cost, performance, and schedule criteria will be met, then the system is said to be "Affordable."

If, over time, some circumstances change (changes in stakeholder requirements, system definition understanding, interface demands, technology maturity, risk exposure, technical measures, or identified defects & errors, or perhaps programmatic influences including impacts of resources, staffing, or budget cuts) the system affordability changes. The first figure reflects the cost trend if confidence is held constant. In contrast, the second figure represents the reverse, the confidence trend if the cost is held constant.

The first figure illustrates the Affordability Leading Indicator as the cumulative expected cost trend at a fixed confidence. At several points during the project, the cost is re-estimated, at this fixed confidence level. The second figure illustrates the same data as a confidence trend if a fixed expected cost is maintained. These figures describe a project that initially has a high confidence at time T1 that costs are low.

Now, suppose the customer asks the supplier to determine the effect of a certain set of changes to the requirements. By time T2, the supplier determines new affordability confidence distributions. If the customer is interested in a fixed confidence, the corresponding cost might increase, or if the original baseline costs were fixed, the corresponding confidence value that the project can be executed at that price would decrease. Together, these might imply that given the requirements change there is more uncertainty.

Under these circumstances there should be a conversation between the supplier and the customer to determine the course of action. Hypothetically, the customer might accept the higher cost baseline, at the original confidence level. On the other hand, if the baseline cost must be held constant and the customer can not accept the reduced confidence level, other alternatives must be considered.

Obviously, under these circumstances further management action is required. Likely, an analysis to determine the root cause of the reduction in affordability confidence would like be initiated. Perhaps, the performance specification was allowed to "creep" without appropriate oversight; a performance justification challenge could be imposed on both the customer and supplier's technical teams. Hypothetically, through decisions reach by project management and the customer at time T3 some certainty has returned confidence improved and the expected cost has gone down.

This example demonstrates one approach to calculate affordability. Another approach is to express change in affordability as a rate change from the baseline:

$$\text{Change in Affordability} = 100\% \left(\frac{\begin{bmatrix}\text{Baseline Effort} \\ \text{and related confidence}\end{bmatrix} + \begin{bmatrix}\sum \text{Estimated Effort}_{\textit{(Impact of Changes)}} \\ \text{and related confidence}\end{bmatrix}}{\begin{bmatrix}\text{Baseline Effort} \\ \text{and related confidence}\end{bmatrix}} \right)$$

Projected costs can be measured in a number of ways – dollars, labor hours, etc. – depending on the situation. Projected schedule (elapsed time to implement a system that meets the needs can be measured as days, weeks, months, years, etc.) The needs might be quantified and measured as software size, number of requirements, etc. Accordingly, affordability indicator is somewhat independent of the technique used to determine it.

3.16.1 System Affordability Trend Specification

System Affordability	
Information Need Description	
Information Need	• Understand the balance between performance, cost, and schedule as well as the associated confidence or risk • Understand whether the intended system –at the development stage or phase of interest – is affordable, i.e., the current confidence that needs can be met at a stated cost/schedule • Understand which aspects of the system are driving affordability – system attributes, risk tolerance, etc. • Understand the trend regarding affordability (including which aspects of the system are driving affordability: performance (system attributes), cost, and schedule as well as the associated confidence or risk)
Information Category	1. Resources and Cost – The measurable System cost projection 2. Schedule and Progress – with respect to Milestone Dates and Product Schedule 3. Risk or Confidence – an understanding of the estimated risk or confidence in cost or schedule projections
Measurable Concept and Leading Insight	
Measurable Concept	Is the SE effort progressing towards a system that is affordable for the stakeholders?
Leading Insight Provided	Indicates whether the emerging system is affordable Indicates what factors might be driving affordability
Base Measure Specification	
Base Measures	1. Baselined Cost (with associated confidence) 2. Planned Cost (with associated confidence) * 3. Baseline Effort (with associated confidence) 4. Planned Effort (with associated confidence) * 5. Baseline Schedule and associated confidence 6. Planned Schedule (with associated confidence) * * Based on estimated impact of changes in stakeholder requirements, system definition understanding, interface demands, technology maturity, risk exposure, technical measures, or identified defects & errors, or perhaps programmatics including impacts of resources, and staffing
Measurement Methods	1. Record Baselined Costs 2. Record Planned Costs 3. Record Baselined Effort 4. Record Planned Effort 5. Record Baselined Schedule 6. Record Planned Schedule

System Affordability	
Unit of Measurement	1. Dollars with associated confidence estimate * 2. Dollars with associated confidence estimate * 3. Labor Hours, Weeks, Months, etc. with associated confidence estimate * 4. Labor Hours, Weeks, Months, etc. with associated confidence estimate * 5. Calendar Weeks, Months, or Years with associated confidence estimate * 6. Calendar Weeks, Months, or Years with associated confidence estimate * * Confidence for each is a unitless percentage
Entities and Attributes	
Relevant Entities	• System Cost/Effort/Schedule Forecast • Changes in Technical Performance or Programmatics
Attributes	• Cost (in dollars) with associated confidence estimate • Effort (in labor hours) with associated confidence estimate • Schedule (in weeks/months/years) with associated confidence estimate • Impact of Changes in Technical Performance or Programmatics expressed in Cost/Effort/Schedule with associated confidence estimate
Derived Measure Specification	
Derived Measure	1. Cost (in dollars) with associated confidence estimate 2. Effort (in labor hours) with associated confidence estimate 3. Schedule (in weeks/months/years) with associated confidence estimate 4. Impact of Changes in Technical Performance or Programmatics expressed in Cost/Effort/Schedule or Percentage Change with associated confidence estimate
Measurement Function	1. Distribution of planned cost (in $) with estimated confidence 2. Distribution of planned effort (in labor hours) with estimated confidence 3. Distribution of planned schedule (in weeks/months/years) with associated confidence) 4. Distribution of Impact of Changes in Technical Performance or Programmatics expressed in Cost/Effort/Schedule or Percentage Change with associated confidence estimate
Indicator Specification	
Indicator Description and Sample	Line or bar graphs that show trends of Cost/Effort/Schedule or Percentage Change with estimated confidence indicated as variances. Show customer expectations or allowable tolerance as thresholds. Show key events along the time axis of the graphs. 1. Distribution of planned cost curve (in dollars) with associated confidence estimate vs. time or event 2. Distribution of planned effort curve (in labor hours) with associated confidence estimate vs. time or event 3. Distribution of planned schedule curve (in weeks/months/years) with associated confidence estimate vs. time or event 4. Distribution of Impact of Changes in Technical Performance or Programmatics expressed in Cost/Effort/Schedule or Percentage Change with associated confidence estimate vs. time or event

System Affordability	
Thresholds and Outliers	Is the confidence in meeting the allocated budget high enough? If not, then do something.
Decision Criteria	Determine what is driving either Cost/Effort/Schedule or confidence and make changes to reach an acceptable level of affordability.
Indicator Interpretation	• Use this indicator to assess system affordability and track system affordability as the system evolves (e.g., concept, design, implantation…) and matures. • Use this indicator to assess system affordability as stakeholder needs and constraints change.
Additional Information	
Related Processes	Stakeholder Requirements, Requirements Analysis, Architectural Design
Assumptions	• Target cost and schedule budgets can be established • Customer sensitivities and thresholds can be estimated • Required confidence estimates can be generated • Customer "needs" and "changes" can be transformed into estimates of system costs, Systems Engineering effort, or project schedule
Additional Analysis Guidance	• Affordability criteria maybe different at different phases or stages of the process • Probably requires the use of parametric cost estimation analysis methods and tools to generate baseline and subsequent cost/effort/schedule with associated confidence • Both top-down and bottoms-up analysis is useful • Historical data associated with budgets, needs, and level of detail in analysis changes is needed
Implementation Considerations	• The use of a cost modeling team (or at least a single point of contact for cost modeling) helps to ensure consistency of analysis • A joint team (shoulder-to-shoulder process) might also help to establish credibility in data • The use of well-accepted tools will also help to establish data credibility
User of Information	• Program/Project Manager • Customer / Stakeholders • Chief Systems Engineer • Product Managers • Designers
Data Collection Procedure	• See Appendix F
Data Analysis Procedure	• See Appendix F

3.17 Architecture Trends

This indicator is used to evaluate the progress that an engineering team is making towards developing a comprehensive system architecture. Good system architecture is critical in conforming to overall technical strategies, being able to trace requirements, managing interfaces to other systems, developing and measuring performance metrics, and in designing and evaluating the system.

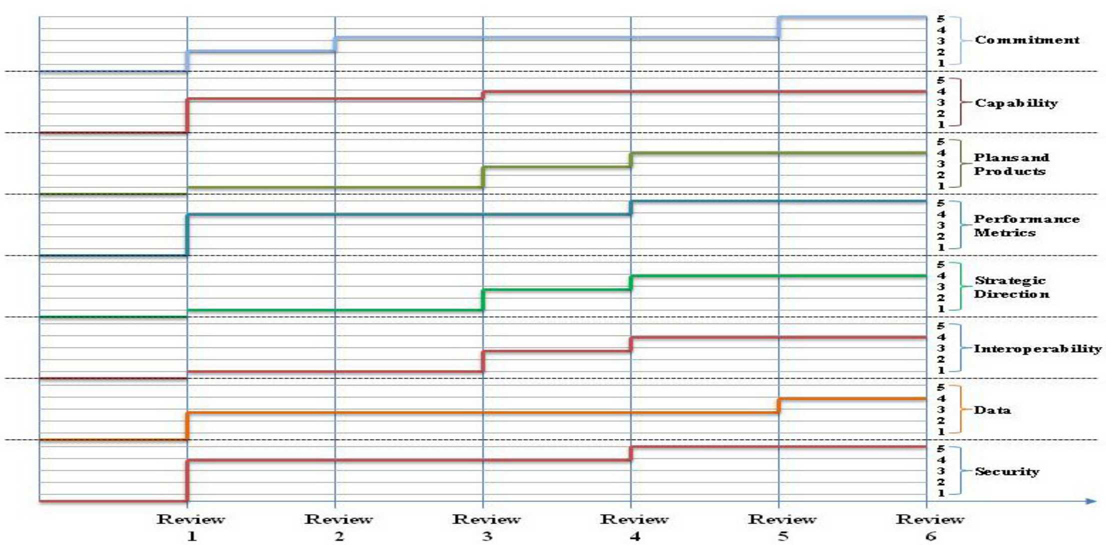
If architecture trends are favorable, there is an increased likelihood that a successful system that fulfills its requirements and is developed according to cost and budget constraints will be produced. In addition, it is also likely that the effectiveness and efficiency of the enterprise itself will improve. This includes innovations in the structure of an organization, the centralization or federation of processes, and the quality and timeliness of information transmission and transformation.

System architects work with leadership, subject matter experts, and all stakeholders to build an integrated view of a system's structure, strategy, processes, and information assets. Architecture links the needed mission capabilities, organizational strategy, processes and documents them from multiple viewpoints that show how the current system and future needs of an organization will be met in an efficient, sustainable, agile, and adaptable manner.

This indicator provides an understanding of the level of organizational strategy and commitment to architectural development, what architectural processes are in place, what skill set the architects have, the degree to which plans (and the mechanisms by which progress against them) are in place, and future needs for skills, methods, and tools. Also addressed are critical issues that can be addressed by architecture in areas such as interoperability, data, and security.

If architectures are assessed in a consistent manner across an enterprise, their assessments can be compared. Each project's assessment could be compared at various points of the project life cycle, to gain further insight of project and system performance. For example, if the results of a project's architecture assessment are compared a group of similar projects further into their own development lifecycle, and it is found to be lower than the norm, management may be legitimately concerned that this project will struggle to be successful. Future research is required to provide more accurate thresholds regarding how large a discrepancy must be to be considered significant.

Here is an example figure of Base Measure scores assessed at various project review points.



Architecture Trend Base Measures					
Score-->	**1**	**2**	**3**	**4**	**5**
Base Measure 1: **Commitment**	• Adequate resources exist	• Responsibility for directing, overseeing, and approving the architecture has been assigned	• Written and approved organization policy exists for architecture development	• Written and approved organization policy exists for architecture maintenance	• Written and approved organization policy exists for IT investment compliance with architecture
Base Measure 2: **Capability**	• A chief architect has been appointed • An office responsible for architecture development and maintenance has been established • An architecture review board exists at the project level	• A formally defined and documented architecture process exists for the organization • A formal architecture training projects exists	• Architecture products and management processes undergo independent verification and validation • There are certified architects on the project • An architecture review board exists at the business unit level	• A formal process exists and is followed to manage architecture change	• Architecture is an integral component of the investment management process • An architecture review board exists at the enterprise level
Base Measure 3: **Plans and Products**	• Architecture is being developed using a framework, methodology, and automated tool • Architecture plans address the "as-is" and "to-be" architecture in terms of business, performance, information/data, application/service, and technology	• Key stakeholder business drivers are documented • The architecture process incorporates the use of domain-specific reference models • The architecture process defines a minimum set of architecture artifacts	• Cognizant organization or individual has approved the architecture plans and products • Process for identifying, managing, and closing gaps between "as-is" and "to-be" is well-documented	• Architecture products are periodically updated • Investments comply with architecture	• The architecture demonstrates the relationships between the "as-is," transition, and "to-be," to investment planning and execution
Base Measure 4: **Performance Metrics**		• Architecture plans call for developing metrics for measuring progress	• Progress against architecture plans is measured and reported	• Compliance with architecture is measured and reported	• Detailed performance measures are defined and linked to the service and technical portions of the architecture
Base Measure 5: **Strategic Direction**	• Architecture demonstrates "front office" and stakeholder buy-in is documented. • Architecture demonstrates management structure and control is established.	• Architecture defines architectural processes • There is a baseline architecture	• Architecture defines a "to-be" (target) architecture • Architecture defines change and risk management strategy or approach	• Architecture defines a transition and sequencing strategy and plan • Architecture defines a communications strategy	• Architecture demonstrates application of the architecture for purposes of creating and maintaining investment projects • Architecture demonstrates an implemented process for managing changes and updates to the architecture
Base Measure 6: **Interoperability**	• Interoperability standards are defined conceptually (patterns, web services, etc.)	• Interoperability standards are defined at the business function level and are aligned to organizational reference models	• Interoperability standards are described through patterns and are related to business functions	• Business functions are aligned to components and services at the enterprise level	• Interoperability and sharing of information is one of the backbones of the target architecture
Base Measure 7: **Data**			• Data architecture is only broadly defined	• Data relationships and interdependencies are defined at a conceptual level	• A common and well-defined approach to integrating data with business processes and mission priorities has been established
Base Measure 8: **Security**		• Security standards are conceptually defined	• Security standards align to a technical reference model	• Security standards are tightly defined and are presented as part of transition planning	• Security standards are tightly defined and are presented as part of investment planning

Chart format suggested by Enterprise Architecture Management Maturity Framework. The first Version 1.0 the Enterprise Architecture Management Maturity Framework was introduced in U.S. General Accounting Office, *Information Technology: Enterprise Architecture Use across the Federal Government Can Be Improved*, GAO-02-6 (Washington, D.C.: Feb.19, 2002).

3.17.1 Architecture Trend Specification

Architecture	
Information Need Description	
Information Need	Evaluates the maturity of an organization with regards to implementation and deployment of an architecture process that is based on an accepted set of industry standards and guidelines
Information Category	• Product Quality • Process Performance • Technology Effectiveness • Customer Satisfaction
Measurable Concept and Leading Insight	
Measurable Concept	• Is the process definition based on industry accepted standards? • Is SE using a defined architecture process through the leadership of certified architects? • Do the architecture work products conform to an industry accepted set of standards?
Leading Insight Provided	• Indicates whether the organization has an architectural process that will assist in maturing the system design • Indicates whether the organization has the architectural skill set in order to execute an architectural process • May indicate future need for different level or type of resources / skills • Indicates whether the system definition is maturing • Indicates schedule and cost growth risk
Base Measure Specification	
Base Measures	1. Commitment 2. Capability 3. Plans and Products 4. Performance Metrics 5. Strategic Direction 6. Interfaces and Interoperability 7. Data 8. Security
Measurement Methods	Self-assessment or independent appraisal
Unit of Measurement	Each Base Measure has an associated unitless level.
Entities and Attributes	
Relevant Entities	• Assessment levels
Attributes	• Assessor contact information • Time Interval (e.g., date, time, monthly, quarterly, phase, etc.) • Objective evidence that support the assessment levels selected • Objective evidence meta-data • Associated attributes (e.g., status, maturity - identified and defined, interval, milestone, type, cause, severity, etc.)

Architecture	
Derived Measure Specification	
Derived Measure	1. Number of base measures failing to improve over time 2. Combined base measure scores 3. Certified architects
Measurement Function	1. Number 2. Weighted average 3. Number
Indicator Specification	
Indicator Description and Sample	Line chart depicting base measures at discrete review points in time.
Thresholds and Outliers	Organization-dependent experience is needed to identify the thresholds and outliers based on comparison to historic project and system performances.
Decision Criteria	Investigate and potentially take corrective action when the base measures do not all improve over time. All measures are expected to exceed level 3 by the time that design begins.
Indicator Interpretation	Lack of progress in any base measures over several periods indicates weakness in the architecting process.
Additional Information	
Related Processes	• Technical Risk • Requirements Analysis • Modeling • Design
Assumptions	Self-assessment is performed by experts with adequate breath of experience and proven judgment.
Additional Analysis Guidance	• System architects must work with leadership, subject matter experts, and stakeholders to build an integrated view of a system's structure, strategy, processes, and information assets to perform the assessment. • Assessment experience will aid in applying the measures in a consistent manner. • Singular assessors are to be avoided whenever possible.
Implementation Considerations	• Record the metadata and examples of objective evidence that supports the base measure level selected. (This might include architecture views, and products, security standards, interface standards, etc.) These data help in recreating or reevaluating the assessments during later project phases.
User of Information	1. Program/Project Manager 2. Chief Systems Engineer 3. Chief Architect 4. Process Lead 5. Architecture Review Board
Data Collection Procedure	See Appendix F
Data Analysis Procedure	See Appendix F

3.18 Schedule and Cost Pressure

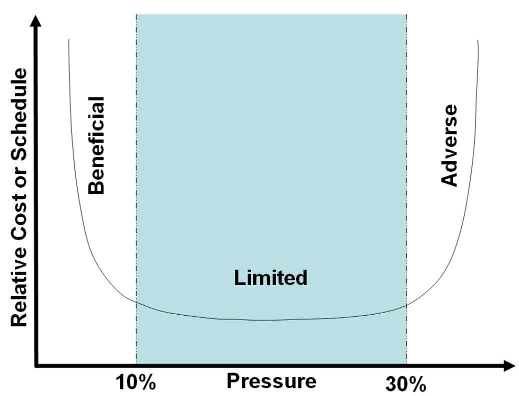
Schedule pressure and cost pressure are generally recognized, at least intuitively, to have an impact on project schedule and cost performance. Some recently published research quantifies the possible effects of "schedule pressure" (SP) and "budget pressure" (BP) and defines measures for them.[7] These measures are defined as relative values, percentage differences between project estimates and contracted values. They have been found to effect performance. Some pressure helps, but too much pressure does not. *While strictly speaking these pressure leading indicators have yet to be widely verified, they intuitively have potential promise.*

More mathematically accurate perhaps, without asserting causality, actual project schedule and cost performance have been found to be closely associated with the values of these measures (BP and SP).

These measures are:

Budget Pressure = ((Project Team Planned Cost)-(Negotiated Contracted Budget))/(Project Team Planned Cost)
and
Schedule Pressure = ((Project Team Planned Schedule)-(Negotiated Contracted Schedule))/(Project Team Planned Schedule)

Research[7] has shown that moderate pressure can have a beneficial performance effect, i.e., moderate pressure may actually cause, or at least be associated with, somewhat reduced resource expenditure and somewhat expedited performance (reduction in period of performance) relative to the initial estimate by the project organization. Intermediate pressure was found to have relatively limited effect, either positive or negative. Excessive pressure caused either an increased period of performance or increased resource expenditure or both. Even though the cited researcher stated reservations based on their limited data collection, years of data collection and analysis have shown schedule compression has significant negative impacts on a project. This was seen in the COCOMO data analysis. Typically, too much schedule compression results in a higher defect rate, which in turn impacts cost, schedule and quality.



Schedule and Cost Pressure. This notional figure represents the relationship between potential relative attained costs or attained schedule and cost or schedule pressure, respectively. For the organizations studied, the researchers defined the pressure boundaries, as depicted. These regional boundary values must be determined based on the experience of the specific organization. The regions designated beneficial, limited, and adverse indicate the effect on cost or schedule anticipated as a function of this pressure. Relationships for cost and schedule vs. pressure, such as that depicted in the figure, would likely be based on the experience of the organization that executes the project. The pressure values (shown on the horizontal axis) would be computed as described below. These values are the leading indicators of "goodness" or "trouble" based on what region into which they fall.

The use of Budget Pressures might be made in comparison to several costs or prices. As examples, one could consider the comparison to the Price-to-Win during the pre-proposal stage of the acquisition life-cycle or the Unit-Production-Cost in the manufacturing development and production stages of the acquisition life-cycle. Price-to-Win is typically set by Marketing or Business Development personnel. Unit-

[7] Impact of Budget and Schedule Pressure on Software Development Cycle Time and Effort by Ning Nan and Donald E. Harter; IEEE Transactions on Software Engineering, Vol. 36, No.5, September/October, 2009, pg. 624.

Production-Cost is typically set by Program Management personnel. Business Development naturally considers the price likely to be bid by the competition and what it takes to win the program. Program Management may set a price so that the recurring cost of a product is optimized at a determined production quantity. In each case, the goal is to balance product capability with cost in terms of value to the customer; this is classic affordability.

For these cases, the Budget Pressure measure might be redefined as:

Budget Pressure = ((Project Team Planned Cost)-(Price-to-Win))/(Project Team Planned Cost)

or

Budget Pressure = ((Project Team Planned Unit Production Cost)-(Target Unit Production Cost))/(Project Team Planned Unit Production Cost)

Furthermore, Schedule Pressure could be redefined in a similar manner. The level of pressure, the value of these measures, can indicate a great deal of tension between Engineering and Business Development, tension based on the different likelihood of success, motivations, drivers, and focus. Accordingly, the schedule and cost pressure percentages should be tracked in the project risk register. Furthermore, these pressures relate to the system affordability in that schedule and cost risk are part of the definition of the affordability leading indicator.

These measures can serve as Leading Indicators of potentially adverse project performance with respect to schedule or cost or both. They can serve as "pre-yellow" and "pre-red" indicators of project schedule or cost performance. (The "yellow" color suggests that the project could trend toward an adverse performance; the "red" color suggests that the project is likely to exhibit adverse performance.) Thus, the indicators could provide early warning to proposal and Program Managers to expect their projects to go to yellow or red before such a transition actually happens. Accordingly, rather than waiting for a project to turn yellow or even sometimes turn directly red, management could act on this indication as soon as the project is initiated. Based on adverse values of the pressure leading indicators, management could possibly take mitigating action. If the project remains "green" meaning that the project will continue to perform well, that it is on target with respect to cost and schedule objectives.

Project data can be collected to provide baselines for the values of the BP and SP indicators, such that an organization "should know what to expect." That is, management should be able to anticipate adverse project performance, in terms of schedule and cost, based on these indicators. For example, management should understand both probability for a range of possible deviations from contracted for values and when such project deviations might occur. Data on other associations, such as the performing organization's estimate of risk (that is, estimated probability of exceeding their estimate), can be developed as well.

Intuitively, it would seem that a best practice would be to apply moderate pressure throughout the development life-cycle. The "pressure" cost and schedule leading indicators could be computed at various times during the course of the project. Thence, knowledge obtained from indicators could be applied incrementally. For example, one could compare the schedule or cost to the most immediate baseline throughout the development life-cycle. They could be re-evaluated periodically and/or when a change in project business environment materially warrants. Although data is not currently available to evaluate this hypothesis, it appears reasonable.

The BP and the SP leading indicators should be found to be closely associated, i.e., be highly correlated with, the Schedule-Performance-Indicator (SPI) and Cost-Performance Indicator (CPI). Although the BP and the SP indicators are top-level or project level indicators, they can serve as prompts for the examination of budgets and expected task performance periods of performance at lower levels of detail.

3.18.1 Schedule and Cost Pressure Leading Indicators Specification

Schedule and Cost Pressure	
Information Need Description	
Information Need	Understand the impact of schedule and cost challenges on carrying out a project.
Information Category	1. Schedule and Progress - System Schedule 2. Resources and Cost - System Cost 3. Risk or Confidence – an understanding of the risk or confidence in cost or schedule projections
Measurable Concept and Leading Insight	
Measurable Concept	To what extent do schedule and cost challenges impact the actual execution of a project?
Leading Insight Provided	Indicates whether the project performance can be adversely effected by efforts to meet customer expectations. Indicates whether the project personnel dynamic can be adversely effected by differences in approaches to meeting competing customer expectations. • Provides early detection or prediction of problems requiring management attention • Allows early action to be taken to address project risks
Base Measure Specification	
Base Measures	1. Planned Cost 2. Planned Schedule 3. Actual Costs 4. Actual Schedule 5. Negotiated, contracted, or targeted costs and schedules
Measurement Methods	1. Record Planned Costs 2. Record Planned Effort 3. Record Actual Costs 4. Record Actual Effort 5. Record negotiated, contracted, or targeted costs and schedules
Unit of Measurement	• Dollars or Time (months, weeks, days)
Entities and Attributes	
Relevant Entities	• System Cost/Effort/Schedule Forecast
Attributes	• Cost (in dollars) with associated confidence estimate • Effort (in labor hours) with associated confidence estimate • Schedule (in weeks/months/years) with associated confidence estimate
Derived Measure Specification	
Derived Measure	1. Schedule Pressure 2. Budget Pressure
Measurement Function	1. Schedule Pressure = ((Project Team Planned Schedule)-(Negotiated Contracted Schedule))/(Project Team-Planned-Schedule) 2. Budget Pressure = ((Project Team Planned Cost)-(Price to Win))/(Project Team Planned Cost)

Schedule and Cost Pressure	
Indicator Specification	
Indicator Description and Sample	• Highlighting the potential for unintended schedule or cost consequences as a result of schedule and cost pressure. • Trends graphs/charts of calculated pressure over time or at project events
Thresholds and Outliers	The effect regions designated beneficial, limited, and adverse indicate the effect of moderate, intermediate, or excessive pressure. The notional regional boundary values below represent the relationships found by the researchers cited. • Moderate pressure (less than 10%) = Beneficial Effect • Intermediate pressure (10%-30%) = Limited Effect • Excessive pressure (greater than 30%) = Adverse Effect Regional boundary values and relationships for schedule and cost vs. pressure should be based on organizational experience.
Decision Criteria	Investigate and potentially take corrective action when the values of exceed the threshold bands or acceptable risk
Indicator Interpretation	• Increased pressure indicates that risk is increasing and that the risk needs to be managed
Additional Information	
Related Processes	System Affordability Analysis and Earned Value Management
Assumptions	• Target cost and schedule budgets can be established • Customer sensitivities are understood • Effect regions thresholds can be estimated • Customer "needs" and "changes" can be transformed into estimates of system costs, Systems Engineering effort, or project schedule • Historical organizational data associating project schedule and budget challenges with project results in terms of schedule and budget are available to establish correlation relationships for schedule and cost vs. pressure and define regional boundary values based on organizational experience
Additional Analysis Guidance	Project results can not be predicted or explained in terms of schedule and budget pressure only. The indication of adverse pressure or pressure trends will likely initiate additional analysis and measurement of a project status.
Implementation Considerations	While historical organizational data is strongly recommended, the use of the notional regional boundary values cited above is suggested in the absence of organizational experience.
User Of The Data	• Chief Systems Engineer • Product Manager • Business Development Manager
Data Collection Procedures	• See Appendix F
Data Analysis Procedures	• See Appendix F

4 IMPLEMENTATION CONSIDERATIONS

When implementing leading indicators, it is important to consider how they are created, used and validated for managing systems engineering activities. Because leading indicators are forward-looking in nature, it is important to develop an understanding of the underlying processes, their previous/historic performance, as well as the extent to which this behavior indicates future performance. In addition, understanding the time frame and strength of a leading indicator is needed to ensure that implementation results in accurate future predictions.

Of course, the future is always uncertain. So, a leading indicator should be viewed as a probabilistic measure. That is, the value of the future state is not a certainty, given the value of the current state; the indicator only suggests what might happen based on past experience. Based on this past experience, we construct predictive relationships or mathematical models. For example, it is possible to predict software defect discovery during system integration and test from the actuals at earlier time periods. We can also use a similar model to predict (i.e., latent) defect discovery post-delivery. Such model-based predictions are not exact, but are probabilistic. We might use such models to predict the expected value, the average, the 90% confidence value, or other single-value prediction.

However to use a leading indicator effectively, we also need to be able to differentiate between good and bad values. So, we need to have expectations for the value of the indicator as well as value thresholds beyond which action is warranted. It is best to establish thresholds for both risk handling actions (the value at which the risk is no longer acceptable, although not yet a problem) and corrective actions (the value at which the parameter indicates a negative impact on the project or system). Unsurprisingly, information needs and/or the sequence of expected or threshold values is likely to be different at different points in time or project phase; so too would the expected action perhaps based on the criticality of the project phase. For example, consider a plan for effort expenditure versus time. A significant departure from the plan, exceeding some threshold value could be an indicator of impending budget trouble. Exceeding the threshold could imply that the appropriate initial action is to simply get more data. On the other hand, the cost or effort indicator, and others, might trigger a root-cause analysis that could identify the initiating source for the departure from desired behavior.

In addition, Appendix A presents a thorough discussion of the leading indicators implementation methodology employed by the Naval Air Systems Command's Systems Engineering Development and Implementation Center.

Finally, any measurement program must account for measure dysfunction or the Hawthorne effect. The Hawthorne effect is where subjects of a study (or measurement program) improve an aspect of their behavior simply in response to the fact that they are being studied and not in response to any particular study imposed change. This guide does not address this well known measure issue and others chiefly because there is a great research available to help mitigate or account for subject bias and measure dysfunction. Measurement professionals can not guarantee that projects will not "game" their measurement, but any metrics program must have "checks" in place for this type of occurrence.

4.1 Evaluating the Cost-Benefit of Leading Indicators

A significant consideration for implementation of any measurement program, including these systems engineering leading indicators, is the associated cost benefit analysis; Is the benefit (knowledge) worth the cost? It is given that some measurement effort is necessary to understand and describe the performance of a business and its projects. Furthermore, without embarking on a return on investment study and acknowledging the importance of Systems Engineering to a project's success, intuitively measurement of Systems Engineering is warranted.

That said, any measurement program must be managed to ensure its cost effectiveness. A business must ensure that the cost of each measurement returns a real information value. Accordingly, serious thought most be given to the selection of measures to be taken and the insight they provide.

How can a leading indicators be implemented most cost effectively? Plainly some measurable concepts provide a basic understanding of a project. These include measures of size, cost, schedule, quality, complexity, and technical performance; these have utility for leading indicators. Leading indicators are often composed of a combination of base measures associated with these concepts. A measurement project that leverages the most widely used base measures can limit implementation costs and effort.

For example, the number of Requirements is the most widely base measure in this guide. Counting the number of Requirements is commonly used to judge the size of the Systems Engineering effort of a project. Regarding size, one could also measure Pages of Specifications, Interfaces, Algorithms, or Operational Scenarios. The general relationship between each of these measures and Systems Engineering effort has been studied and is understood. Each of these measures provides a sense of project size, but determining them may be more difficult than the method of counting the number of requirements in a database.

Some concepts are difficult to measure, Complexity for example. Consequently, some measurements serve as surrogates or substitutes for other measures. Recognizing that large projects are inherently more complex to manage, requirements are often considered a cost effective surrogate for other complexity measures. Furthermore, the number of Requirements, as a measurement of size, is also often used to normalize several of the leading indicators recommended in this document. As a result, measurement of the number of Requirements is considered as an essential base measure.

While the notions of measurement efficiency, surrogacy, and normalization are important, they must be considered in balance with the need for visibility across a project (breath of project understanding and insight). The benefit of having broad understanding of Systems Engineering is that it provides better insight of the true health and probable future of a project.

Is incremental additional business value, additional project or systems engineering insight or reduction of risk, sufficiently worth the additional cost? If the costs of implementation are equal, which measurement is more effective? Only a purposeful tradeoff analysis assessing the incremental value to the business or project can determine the correct answer.

Table 2 below may support this tradeoff analysis. This table depicts the relationships/affinity between the current set of Systems Engineering Leading Indicators. For example, if you again consider the Requirements Leading Indicator, the table portrays ten (signified parenthetically) other related indicators. The related indicators are denoted by an "X" in the corresponding column across the Requirements Leading Indicator row.

Table 2 – LEADING INDICATOR AFFINITY

	Requirements	System Definition Change Backlog	Interface	Requirements Validation	Requirements Verification	Work Product Approval	Review Action Closure	Risk Exposure	Risk Treatment	Technical Maturity	Technical Measurement	Systems Engineering Staffing & Skills	Process Compliance	Test Completeness	Facility and Equipment Availability	Defect and Error	Algorithm/ Scenario	System Affordability	Architecture	Schedule and Cost Pressure
Requirements (10)		X		X	X	X							X	X		X	X	X	X	
System Definition Change Backlog (3)	X		X			X														
Interface (9)		X		X	X	X			X	X				X		X			X	
Requirements Validation (4)	X		X		X					X										
Requirements Verification (9)	X		X	X		X				X			X	X		X	X			
Work Product Approval (5)	X	X	X		X							X								
Review Action Closure (3)										X	X			X						
Risk Exposure (6)									X		X			X	X			X		X
Risk Treatment (9)			X					X		X	X	X			X	X	X	X		
Technology Maturity (8)			X	X	X		X		X			X	X	X						
Technical Measurement (6)							X	X	X			X		X					X	
Systems Engineering Staffing & Skills (6)						X			X	X	X				X					X
Process Compliance (3)	X				X					X										
Test Completeness (11)	X		X		X		X	X		X	X					X	X		X	X
Facility and Equipment Availability (5)								X	X			X						X		X
Defect and Error (6)	X		X		X				X					X					X	
Algorithm/Scenario (5)	X				X				X					X					X	
System Affordability (5)	X							X	X						X					X
Architecture (6)	X		X								X			X		X	X			
Schedule and Cost Pressure (5)								X				X		X	X			X		

Finally, it is important to note that the guide provides implementation examples. The examples listed are in no way exhaustive. Consequently, associating any artificial importance to the quantity of relationships listed in this guide is not recommended; however the quantity of relationships between leading indicators within your implementation is an important consideration. Given this analysis, a standard recommendation would not likely fit every project or business situation. The best set of leading indicators will obviously be effected be the project complexity and the business's existing measurement infrastructure.

4.2 Leading Indicator Performance

If appropriate, it is wise to employ a measurement to quantify the performance (accuracy and precision) of a leading indicator. This section illustrates the use of such a measurement; we will consider a scenario of its use to quantify the performance of the Affordability Leading Indicator.

The Affordability Leading Indicator is a cumulative probability distribution; the probability that the (actual) cost will be less than or equal to some desired value. Accordingly, the Affordability Leading Indicator is determined at several times during the course of a project, providing several opportunities to judge how well the leading indicator is projecting the cost.

For those Leading Indicators that are defined as probability distributions, we can quantify the performance quality of the indicator using the relative dispersion of the probability. We refer to this measurement as the Estimation Performance Indicator (EPI). The EPI is a statistical measure of the coefficient of variation, equal to the standard deviation divided by the average or expected value. The EPI can be seen as a measure of ignorance or confidence. The greater the value of the EPI, the more uncertain (or ignorant) one is about what the actual value of the measure to which the EPI refers.

In the case of the Affordability Leading Indicator, the EPI indicates the relative "fatness" of the distribution of the cost. As successive estimates are made, the Affordability is computed and the value of the associated EPI should be expected to trend lower over time. The estimates should be more precise. As there is no uncertainty as to the cost when the project is complete, the final EPI should be zero.

Figure 3 shows a time progression of the Affordability Projected Cost Confidence (cumulative cost distribution), for four different times, three during the project (T1, T2, T3) and one at the end of the project (TF). Figure 4, Time Progression of Cost Probabilities, shows four probability frequency functions corresponding to the same four times.

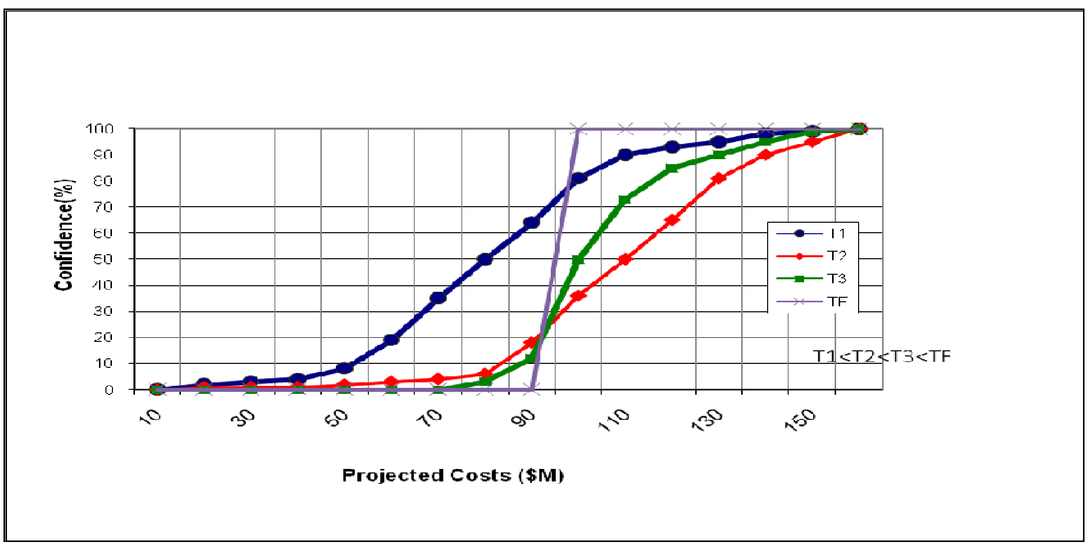


Figure 3 – Time Progression of Projected Cost Confidence

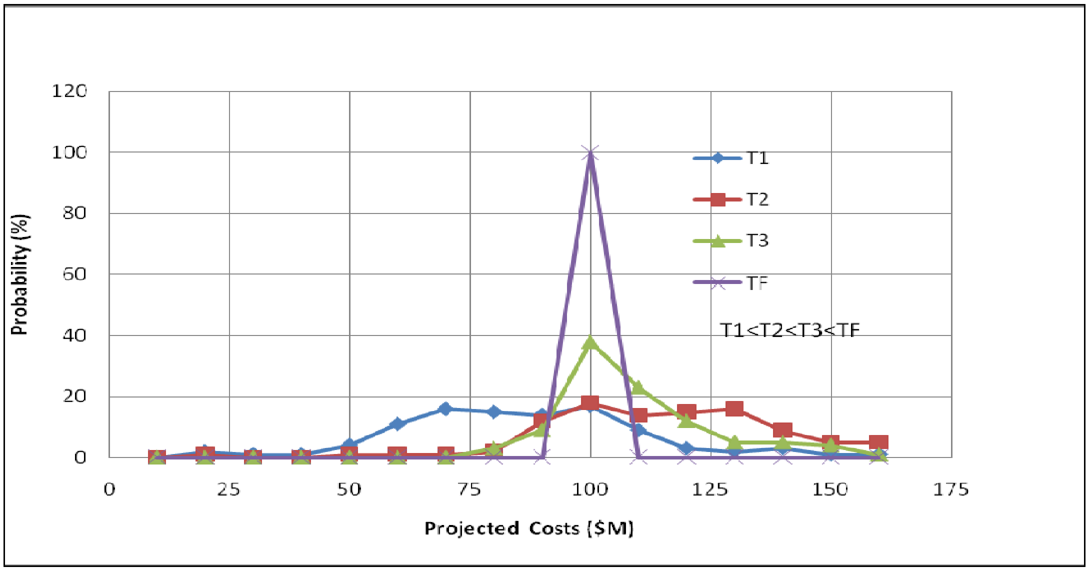


Figure 4 – Time Progression of Cost Probabilities

Table 4 provides the statistics for Affordability at these times. As suggested, the EPI values decrease as the project progresses. The decreasing values indicate that, as better data becomes available the system is being estimated to be "less affordable" (i.e. the original estimate was not accurate), the precision of that estimate itself is increasing. (Notice the variation of the successive average, median, and 90% confidence cost values suggests the risk of attaining the target cost rises from time T1 to time T3.) Thus, the persons doing the estimate have an increasingly better idea of what the project is going to cost at the end of the project, at t=TF. Their uncertainty about the factors driving the cost is being reduced. Consequently, the EPI for time=TF, is of course, zero; as the project has been completed.

Table 3 -Cost Statistics

	Successive Estimate Times→			
Item	T1	T2	T3	TF
Average	85.90	114.70	109.30	100.00
Median	80.00	110.00	100.00	XXXXXXX
90% Confidence	110	140	130	XXXXXXX
StdDev	43.40	43.32	27.95	0.00
EPI*	0.51	0.38	0.26	0.00

** EPI=Estimation Performance Indicator*
(Coefficient of Variation=StdDev/Avg)

It is helpful to quantify the performance of the leading indicators. Measurements, such as the EPI, can increase the confidence in applying leading indicators.

4.3 Use of Composite Indicators

There are a number of leading indicators that are composed of multiple measures or even other indicators. We refer to these as Composite Indicators. Often, the purpose of a composite indicator is to understand a specific characteristic of a project or to enable a specific conversation with a stakeholder.

One such composite leading indicator is System Affordability Trends. The System Affordability Trend was developed, in part, to enable more beneficial conversations between the Systems Engineering Manager, the Program Manager, and perhaps a Customer representative. It was based on thinking represented by the following summarized proposal requirement…

> *…the contractor is to provide an estimated Life Cycle Cost, with uncertainty. The contractor must show the relationship between the uncertainty and the associated potential technical, schedule, and cost risks or impacts. The cost analysis is to be performed from the 10th to 90th confidence level. Finally, the contractor is to continuously monitor the estimated Life Cycle Cost throughout the development phase, contrasting the risk and design trades with cost and schedule sensitivities…*

Some have questioned the "leading-ness" of such an indicator or have suggested that the use of "estimation" limits the integrity of the indicator.

In some cases, for example if indicators are simple aggregations, then perhaps these questions are valid. However in other cases, if the indicator provides a vehicle to have a more beneficial conversation with a customer, in terms that the customer appreciates, such an indicator has value. Also, the use of one indicator does limit the use of other indicators and measurements to support a more detailed discussion. Furthermore, it may be wise to employ more than one leading indicator to derive information about how a project is progressing.

4.4 Indicators vs. SE Activities

Table 2 is a mapping of the SE Leading Indicators to the process activities listed in ISO/IEC 15288, System Life Cycle Processes. The table only includes the system life cycle processes for which there are at least one SE Leading Indicator that maps to one of the activities of that process. The identified relationships in this table reflect the activities in which the leading indicators are most likely to provide useful insight.

Table 4 - LEADING INDICATORS APPLICATION PER ISO/IEC 15288

	Requirements Trends	System Definition Change Backlog Trend	Interface Trends	Requirements Validation Trends	Requirements Verification Trends	Work Product Approval Trends	Review Action Closure Trends	Risk Exposure Trends	Risk Treatment Trends	Technology Maturity Trends	Technical Measurement Trends	Systems Engineering Staffing & Skills Trends	Process Compliance Trends	Test Completeness Trends	Facility and Equipment Availability Trends	Defect/Error Trends	System Affordability Trends	Architecture Trends	Schedule and Cost Pressure
6.3 Project Processes																			
6.3.1 Project Planning Process																			
6.3.1.3.a Define the project																			
6.3.1.3.b Plan the project resources												x			x				x
6.3.1.3.c Plan the project technical and quality management						x	x									x			
6.3.1.3.d Activate the project																			
6.3.2 Project Assessment and Control Process																			
6.3.2.3.a Assess the project						x	x					x	x		x	x			x
6.3.2.3.b Control the project						x	x					x	x		x	x			x
6.3.2.3.c Close the project																			
6.3.3 Decision Management Process																			
6.3.3.3.a Plan and define decisions										x							x		
6.3.3.3.b Analyze the decision information										x							x		
6.3.3.3.c Track the decision										x							x		
6.3.4 Risk Management Process																			
6.3.4.3.a Plan Risk Management																			
6.3.4.3.b Manage Risk Profile																			
6.3.4.3.c Analyze Risks								x											
6.3.4.3.d Treat Risks								x	x										
6.3.4.3.e Monitor Risks								x	x										
6.3.4.3.f Evaluate Risk Management Process								x	x										
6.3.5 Configuration Management Process																			
6.3.5.3.a Plan configuration management																			
6.3.5.3.b Perform configuration management		x																	

Table 4 - LEADING INDICATORS APPLICATION PER ISO/IEC 15288

	Requirements Trends	System Definition Change Backlog Trend	Interface Trends	Requirements Validation Trends	Requirements Verification Trends	Work Product Approval Trends	Review Action Closure Trends	Risk Exposure Trends	Risk Treatment Trends	Technology Maturity Trends	Technical Measurement Trends	Systems Engineering Staffing & Skills Trends	Process Compliance Trends	Test Completeness Trends	Facility and Equipment Availability Trends	Defect/Error Trends	System Affordability Trends	Architecture Trends	Schedule and Cost Pressure
6.3.6 Information Management Process																			
6.3.6.3.a Plan information management																			
6.3.6.3.b Perform information management		x																	
6.4 Technical Processes																			
6.4.1 Stakeholder Requirements Definition Process																			
6.4.1.3.a Elicit Stakeholder Requirements	x																		
6.4.1.3.b Define Stakeholder Requirements	x																x		
6.4.1.3.c Analyze and Maintain Stakeholder Requirements	x	x		x							x						x		
6.4.2 Requirements Analysis Process																			
6.4.2.3.a Define System Requirements	x																x		
6.4.2.3.b Analyze and Maintain System Requirements	x	x		x	x						x						x		
6.4.3 Architectural Design Process																			
6.4.3.3.a Define Architecture			x							x							x	x	
6.4.3.3.b Analyze and Evaluate Architecture			x							x	x						x	x	
6.4.3.3.c Document and Maintain Architecture		x	x															x	
6.4.4 Implementation Process																			
6.4.4.3.a Plan implementation																			
6.4.4.3.b Perform implementation											x								
6.4.5 Integration Process																			
6.4.5.3.a Plan integration														x					
6.4.5.3.b Perform integration											x			x					
6.4.6 Verification Process																			
6.4.6.3.a Plan verification					x									x					

Table 4 - LEADING INDICATORS APPLICATION PER ISO/IEC 15288

	Requirements Trends	System Definition Change Backlog Trend	Interface Trends	Requirements Validation Trends	Requirements Verification Trends	Work Product Approval Trends	Review Action Closure Trends	Risk Exposure Trends	Risk Treatment Trends	Technology Maturity Trends	Technical Measurement Trends	Systems Engineering Staffing & Skills Trends	Process Compliance Trends	Test Completeness Trends	Facility and Equipment Availability Trends	Defect/Error Trends	System Affordability Trends	Architecture Trends	Schedule and Cost Pressure
6.4.6.3.b Perform verification					x									x					
6.4.8 Validation Process																			
6.4.8.3.a Plan validation				x										x					
6.4.8.3.b Perform validation				x										x					
6.4.10 Maintenance Process																			
6.4.10.3.a Plan maintenance										x									
6.4.10.3.b Perform maintenance										x									

4.5 Potential Future Indicators

4.5.1 Complexity Measurement

A set of measures relating complexity was planned for this revision of the leading indicator guide. Complexity has been one of the most-commonly requested indicators. The team that updated the guide researched several approaches to developing a leading indicator related to the measurement of complexity. Unfortunately, several concerns prevented its inclusion at this time.

Primarily, there was lack of clarity and agreement about the following items:

- How to approach creation of a measure that is actually not in place in any of the contributing organizations? The current guide is comprised of measures/indicators that are in use by one of the contributing organizations. Speculative leading indicators are avoided to the greatest extent possible.
- There are so many facets to complexity that there is no one definition - it depends on the situation/perspective. Given the lack of clarity on a common definition for complexity, it is not clear we will know how to have a valid measure of it.
- Generally, the guide has emphasized easily measured items. The resulting complexity indicator would likely be a composite indicator composed from the existing indicators and potentially additional measures; statistical or logical correlation should be established. In contrast, theoretically rigorous items (e.g. information entropy) may not be easily measured.
- Should the measure address the complexity of the system being built (product elements and interfaces, e.g.) or the development project building such a product (tasks and schedules), or perhaps a combination of these approaches?
- How much of the measure should address structure (size, connectivity, and patterns) vs. dynamics (short- and long-term) vs. socio-political complexity?
- How well a model can capture real-world complexity? Measurement is inherently model-driven: Methods (which convert Attributes into Base measures), Functions (which convert Base Measures into Derived Measures), and Models (which convert Derived Measures into Indicators) are each transformations of one model entity into another. The problem is that models are always more simplified than the reality they represent. (Otherwise, what would be the point of the model?) Creating an inherent contradiction or dichotomy; a complexity measure, by definition a less-complex representation than reality, is trying to represent the complexity of reality.
- Recommended usage of the proposed complexity measurement. Although there is adequate evidence that complex items are riskier than items that are less complex, per Ashby's law of requisite variety, a control system that performs a complex task cannot do so without a minimum level of complexity. So how well understood must the complexity measurement be in order to assure that the systems built are "as simple as possible, but no simpler"? (Attributed to Albert Einstein.)

These issues are being addressed currently in a PhD thesis of one of the authors. A complexity related indicator remains a goal of any future revision of this guide.

4.5.2 Other Potential Indicators

The following list includes a number of potential indicators to be considered in a future revision.

- Algorithms/Scenarios - Rate of maturity of the system algorithm and scenario definition against the plan. Additionally, characterizes the stability and completeness of the system algorithm and scenario definition.
- Design Margin - Risk associated with design margins. The probability of non-compliance to design margins could represent significant project risk during development and have operational effectiveness/customer satisfaction impact.
- Organizational Factors – Address the adequacy of the organizational factors to the effectiveness of the Systems Engineering and project success.

Future versions of the guide should address development of additional indicators that address the later stages of the development life-cycle. These may include:

- Test Completeness - Characterizes the completeness of the system test execution against the plan. Additionally, provides insight to the rate of maturity of the system tests and responsiveness of the organization in closing post-test actions.
- Test Effectiveness - Characterizes the usefulness of the system test by measuring the number of errors found in test vs. the number of errors escaping to the customer.
- Manufacturing Readiness Level - Risk associated with preparation for manufacturing. Premature advancement to manufacturing could introduce significant risk during development. Manufacturing Readiness Levels (MRLs) were developed to provide an understanding of manufacturing risk and maturity in a manner similar to Technology Readiness Levels. The use of MRLs to assess manufacturing readiness can foster better decision making, project planning and execution through improved understanding and management of manufacturing risk.
- Integration - Integration execution against plan. Additionally, provides insight to the rate of maturity of the system and responsiveness of the organization in closing integration actions.

5 REFERENCES

1. Report on the Air Force/LAI Workshop on Systems Engineering for Robustness, July 2004, http://lean.mit.edu
2. Interim Defense Acquisition Guide, Dec 2009 Draft, http://akss.dau.mil/dag
3. ISO/IEC 15288, Systems Engineering – System Life Cycle Processes, ISO/IEC Apr 2008
4. Guidance for the Use of Robust Engineering in Air Force Acquisition Programs, Air Force Center for Systems Engineering, January 2004, http://cse.afit.edu
5. ISO/IEC 15939, Systems and Software Engineering - Measurement Process, ISO/IEC 2007
6. Measurement Primer, International Council on Systems Engineering, March 1998, www.incose.org (Revision pending release in early 2010)
7. PSM Guide V4.0c, Practical Software and Systems Measurement, Department of Defense and US Army, October 2000
8. Technical Measurement Guide, Version 1.0, INCOSE & PSM, December 2005 http://www.incose.org and http://www.psmsc.com
9. CMMI® (Measurement and Quantitative Management Process Areas), Version 1.3, August 2006, Software Engineering Institute, http://www.sei.cmu.edu
10. Practical Software Measurement: Objective Information for Decision Makers, Addison-Wesley, 2002
11. "A Framework for Assessing and Improving Enterprise Architecture Management (Version 1.1)," United States General Accounting Office Report GAO-03-584G, April 2003
12. "Guidelines for Enterprise Architecture Assessment Framework," OMB FEA Program Management Office, April 2004
13. "Improving Agency Performance Using Information and Information Technology (Enterprise Architecture Assessment Framework v3.0)," OMB, December 2008
14. O'Brien, Liam, Len Bass, and Paulo Merson, "Quality Attributes and Service-Oriented Architectures," CMU/SEI/2005-TN-014, September 2005
15. Feiler, Peter H., David P. Gluch, and John J. Hudak, "The Architecture Analysis & Design Language (AADL): An Introduction," CMU/SEI-2006-TN-011, February 2006
16. Kazman, Rick, Mark Klein, and Paul Clements, "ATAM: Method for Architecture Evaluation," CMU/SEI-2000-TR-004, August 2000
17. The US DoC IT Architecture Capability Maturity Model (ACMM), The Open Group, undated white paper
18. ESC Enterprise Integration Toolkit, Self-Assessment Checklist–Program Level, undated DRAFT
19. "ESC/EN Guidelines for Leading Indicators," DRAFT v1.0d, 8 January 2008
20. "ESC Enterprise Architecture Process," DRAFT, 24 September 2007
21. Couretas J. M., System Architectures: Legacy Tools/Methods, DoDAF Descriptions and Design Through System Alternative Enumeration, JDMS, Volume 3, Issue 4, October 2006 Pages 227–237
22. Rhodes, D.H., Valerdi, R. and Roedler, G.J., "Systems Engineering Leading Indicators for Assessing Program and Technical Effectiveness," *Systems Engineering*, Vol. 12, No. 1, pp. 21-35, Spring 2009

APPENDIX A – NAVAIR Applied Leading Indicator Implementation

A.1 INTRODUCTION

The methodology and resulting tool set described in this Appendix is an example of research and analysis stemming from the guidance provided in the main body of the Systems Engineering Leading Indicators Guide. The Naval Air Systems Command (NAVAIR, AIR-4.1) Systems Engineering Development and Implementation Center (SEDIC) developed a methodology that can be used to apply the Leading Indicator (LI) guidance and definitions provided in the body of this LI Guide to a set of specific data. This methodology has been successfully applied to a few technical measures commonly used by NAVAIR Acquisition Category (ACAT) I & II aircraft development programs. The resulting tool is referred to as an Applied Leading Indicator (ALI).

These ALI Tools are designed to provide organizations with a quantitative projection of how their various technical performance measures are impacting overall program performance (i.e., cost and schedule). The ALI Tools are designed to provide current and projected program performance. This ALI Tool fills a gap that currently exists between technical measures and overall program performance measures as shown in Figure A-1.

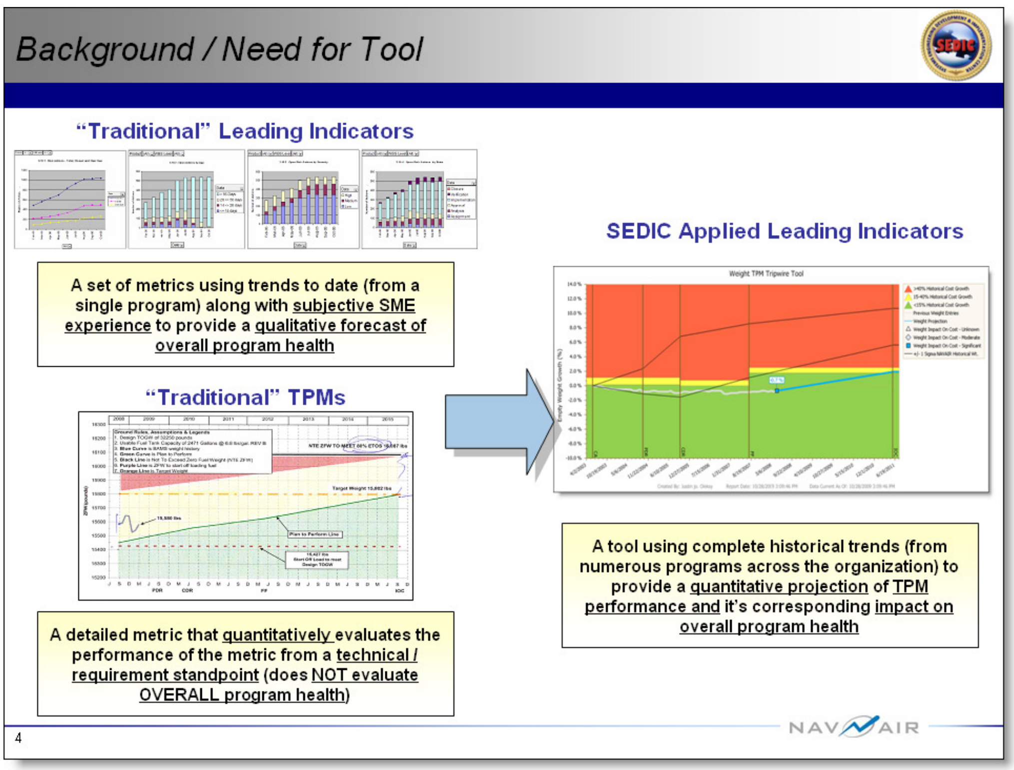


Figure A-1 - ALI Relationship to Existing Performance Measures

This methodology is being provided for public use in anticipation that other organizations will use this to develop their own ALI Tools. As other organizations gain additional experience, collaboration will be encouraged to further refine the methodology used to develop these ALI Tools. The authors urge organizations interested in adopting these tools to contact the NAVAIR SEDIC and LI Working Group personnel listed at the beginning of this Guide.

A.2 PURPOSE

The ALI Tool is intended to provide systems engineers with a prognostic and quantitative evaluation of the "goodness" of technical measures/Technical Performance Measurement (TPM) performance compared to historical NAVAIR program execution. Furthermore, the tool is intended to provide a projection of the effect of the technical measures/TPM on overall program performance (A1)[8].

Overall program performance is known to be influenced by a number of factors. One common representation of program performance is the cost-schedule-technical triangle, as shown in Figure A-2. Notably, the cost and schedule edges are readily measured in terms of dollars and days, respectively. However, it is more difficult to put a gauge on technical achievement due to the disparate dimensions to technical performance.

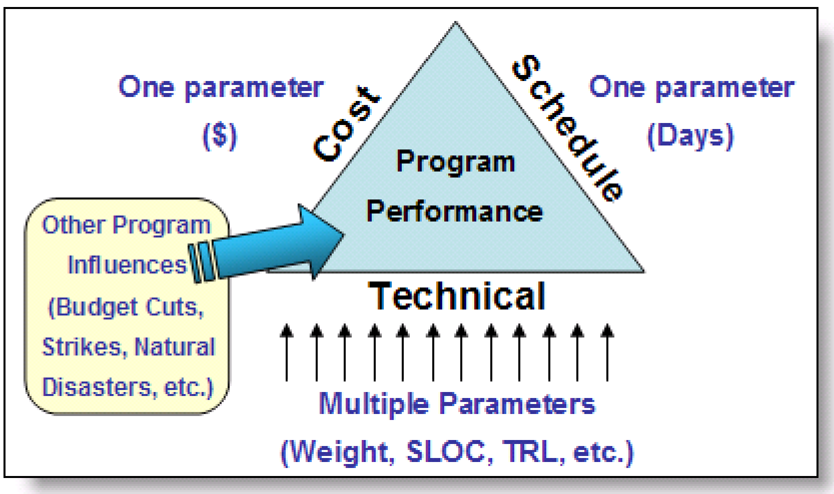


Figure A-2 - Program Performance "Triangle"

A single ALI Tool, developed using this methodology, projects the cost growth or schedule growth based on a single technical parameter out of the various possible technical parameters. As such, the tool has limited accuracy and utility since it does not account for the other technical perspectives. The SEDIC developed a pilot of the ALI Tool based on actual demonstrated program performance of a sample of historical development programs. Currently, this methodology has been applied to NAVAIR ACAT I & II aircraft development program data where program performance is being represented by cost growth defined as the percent overall program cost growth from the original budget. The technical parameter selected was platform empty weight. The resulting ALI Tool is intended to be used as a rough indicator (or "Tripwire") of the health of a program's weight from an overall program cost perspective.

Multiple ALI Tools can be developed using various technical measures/TPMs, incorporating staffing, requirements, Software Lines of Code (SLOC), etc., in addition to weight. Although a single model will never explain all of the variability in cost or schedule performance, multiple ALI Tools can be used together, resulting in improved accuracy and utility. A suite of ALI Tools would provide more comprehensive coverage of the technical edge of the program performance triangle. The SEDIC is currently developing a "suite" of ALI Tools (see Figure A-3) that can be combined to gain a better understanding of overall program performance, and the various technical measures that will impact overall program performance. This will provide programs with a proactive tool to help them identify

[8] For the first iteration of the tool, the SEDIC has elected to use overall program cost as the sole measure of program performance with the intention to add additional measures of program performance in the future (i.e., overall program schedule performance).

program technical risks early so that program resources can be focused to these specific areas. This will assist programs in achieving optimal results from their limited program resources available for risk mitigation, and help improve program success.

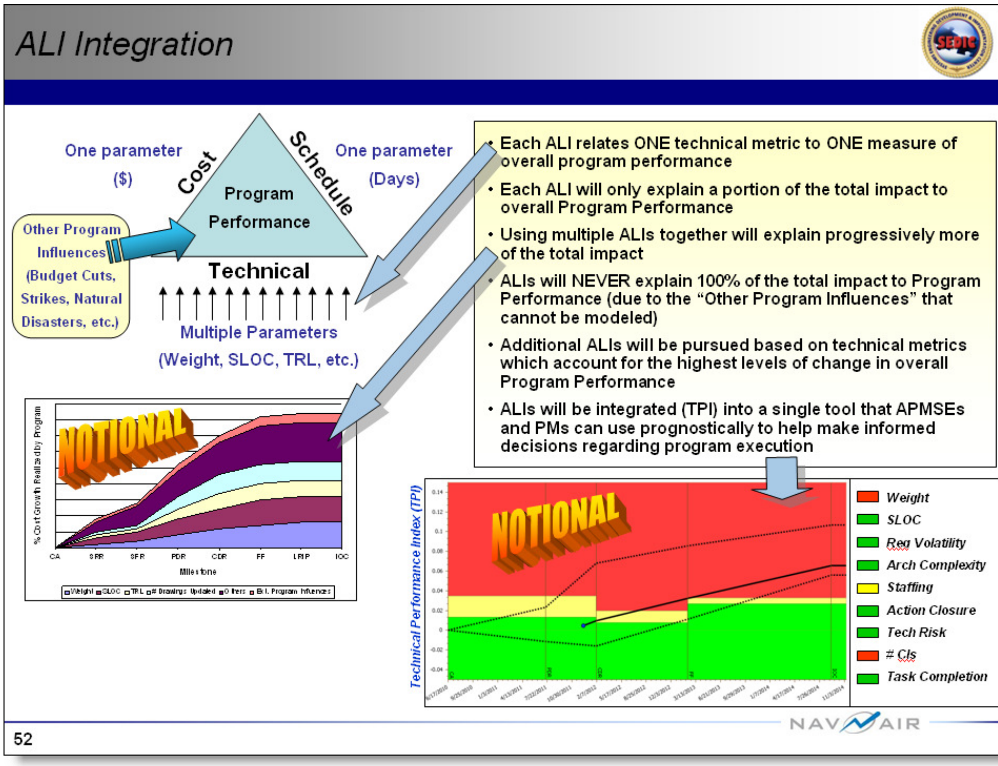


Figure A-3 - SEDIC ALI Vision: Suite of ALI Tools Used in Combination

A.3 METHODOLOGY

The ALI Tool methodology is meant to be an iterative process, with successive development cycles adding to the explanatory power of its predecessor. It should be noted this methodology has been developed as such that it will work with any combination of technical measure and overall program performance measure. The SEDIC has used this methodology on various parameter combinations. In this appendix, % empty weight growth (technical measure) and % overall program cost growth (overall program performance measure) will be used to demonstrate this methodology. However, any organization can use this methodology to develop ALI Tools based on any parameters they choose.

The percent cost growth (%CG) parameter used in this Appendix is defined as the percent cost growth from Original Contractor Budget Baseline (OCBB) versus the current Estimate At Complete (Most Likely) (EAC ML). Additional overall program execution measures are currently being evaluated for use in future tool revisions (i.e., percent schedule growth, probability of Operational Evaluation (OPEVAL) success, etc.).

The first set of parameters pursued by the SEDIC included aircraft weight and overall program cost growth. These parameters were initially pursued given that past research has shown that weight is typically the most significant factor that correlates to aircraft development program cost growth. The statistical relationship between weight growth and cost growth has been thoroughly documented by the RAND Corporation [A1],[A2]. The remainder of this Appendix refers to these parameters (weight growth vs. cost growth) when describing the six-phase methodology used to develop the ALI Tool. Figure A-4 summarizes the six-phase methodology utilized to produce the ALI Tools.

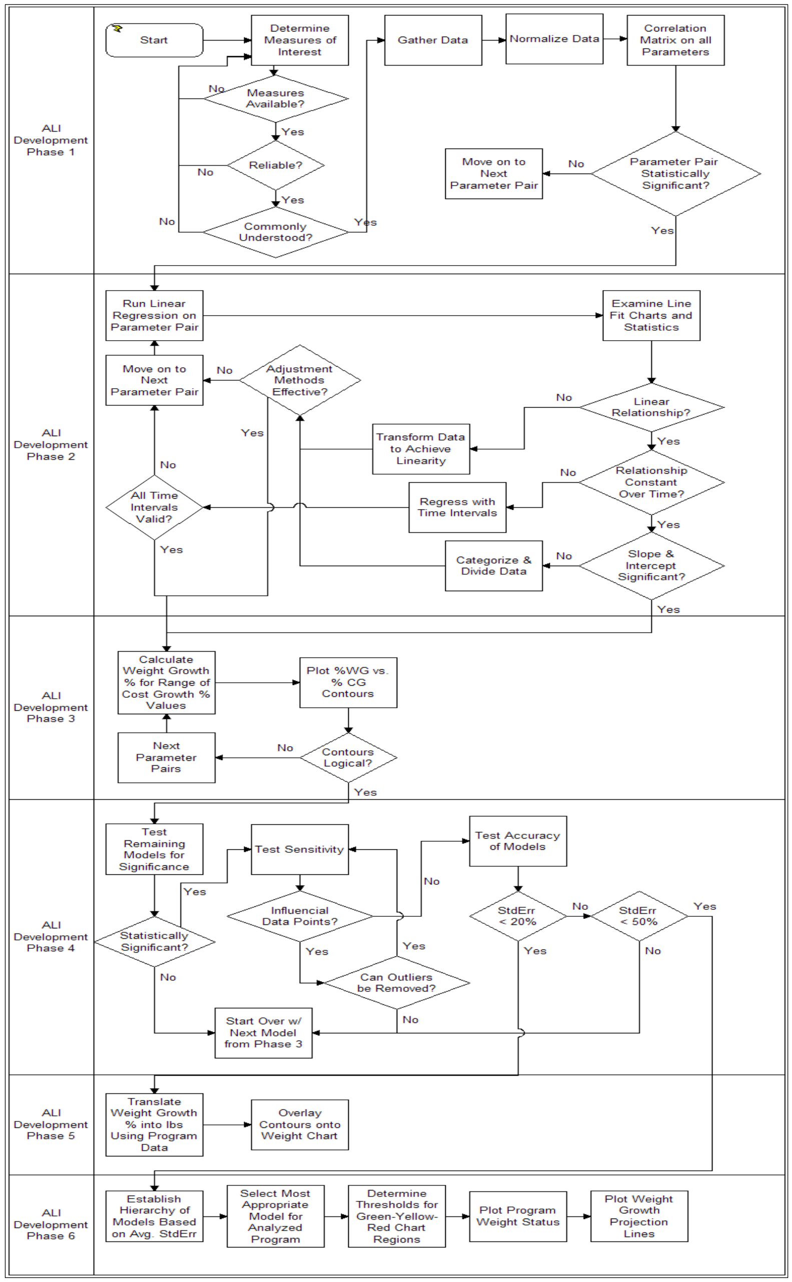


Figure A-4 - ALI Development Methodology Flow Chart

A.3.1 ALI Development Phase 1

Perform global correlation analysis on technical measures versus various program performance (cost/schedule/technical) parameters to identify potentially strong relationships.

The first step in developing an ALI Tool is determining the initial technical measures to pursue based on the utility of each measure as a predictor (or true leading indicator). Some factors to consider are: availability of metric data, understanding of metric across the organization, extent of common usage, accuracy of the metric, etc. As discussed previously, weight was a technical measure that had important implications for a system's operational capabilities, as well as a history of being linked to program cost. This history, coupled with the fact that detailed weight data has been collected and studied by the NAVAIR Systems Engineering Organization, made detailed weight data a logical metric to investigate first.

Gathering the technical measure data and the overall program performance data is no small undertaking. There are many hurdles to the data collecting process that must be overcome. At times, there is little consistency between different programs, including whether or how certain measures are utilized or calculated. Too often, the data is not available in sufficient quantity to support robust statistical analysis. The sample data set must also adequately represent the larger population of programs and cover the full development life cycle if the ALI Tool is to produce usable projections for the entire population of an organization's programs. The NAVAIR Mass Properties Division kept monthly weight status reports for most large NAVAIR programs. The NAVAIR Cost Department also had monthly data for all large cost reimbursable aircraft development contracts. Therefore, enough data was available for the chosen measures to proceed.

In many cases, the technical data is not provided in the format that is best suited for the analysis. Different programs often use the same measures and units of measure, but at different magnitudes. For example, taking the weight TPM, the data is usually kept in terms of pounds or kilograms. While it makes sense to track weight for programs individually with this unit of measure, it becomes problematic when different programs' data are aggregated in a statistical model. For example: with Unmanned Air Systems (UAS) being much smaller and lighter than Fixed Wing Fighter Aircraft, it is not realistic to expect a single statistical model to accurately make predictions for both types of platforms. However, this situation can be remedied by translating the data into dimensionless measures. The weight TPM data collected for the ALI Tool included data indicating the program's status weight, planned weight, and Not-To-Exceed (NTE) weight limit in pounds. Rather than using these measures directly to fit statistical models, the Team computed dimensionless measures such as percent below weight plan (%BP), percent below NTE weight limit (%BNTE), and percent cumulative weight growth from original estimate (%CWG) to serve as the basis for modeling.

A similar concept applies to the overall program health measures such as program cost and schedule. Cost performance measures such as the Cost Performance Index (CPI), Schedule Performance Index (SPI), and %CG are dimensionless, and therefore preferable to measures such as the Variance at Complete (VAC), Estimate at Complete (EAC), and others that are measured in dollars.

As shown in the example in Figure A-5 below, a correlation matrix should be used to perform a correlation analysis[9] on each technical measure versus all possible program performance measures (cost/schedule/technical) In Figure A-5, three sample weight measures and three sample overall cost measures are analyzed.

---

[9] The preferred approach is to use parametric methods when possible. If the assumptions behind a parametric approach are violated, or a strictly linear relationship does not exist between the parameters chosen, the Spearman Rank Order correlation coefficient can be used instead.

CORRELATION MATRIX:

Technical Metric
Performance Metric
Test for significance

	% CWG	% BNTE	% BP	CPI	SPI	% CG
% CWG	1	0.41419	0.68481	0.18595	0.90842	0.80775
% BNTE	0.41419	1	0.81171	0.45719	0.94942	0.08434
% BP	0.68481	0.81171	1	0.00444	0.03593	0.03052
CPI	0.18595	0.45719	0.00444	1	0.17927	0.0849
SPI	0.90842	0.94942	0.03593	0.17927	1	0.32032
% CG	0.80775	0.08434	0.03052	0.0849	0.32032	1

Figure A-5 - Correlation Matrix

The correlation matrix shows the Pearson's R (the Pearson product moment correlation coefficient)[10] for all possible parameter pairs. Each technical metric will be tested for correlation to each performance metric for statistical significance.

Note that the correlation between two technical measures or two performance measures (shown in the white and gray cells in Figure A-5) is not pertinent to the ALI Tool's objectives. Because each iteration of the ALI Tool is looking for a single weight metric that can predict the cost growth on a particular program, the fact that %BNTE and %BP show a high degree of correlation to each other precludes either as a candidate for use in the ALI Tool. However, if a multi-variable model was being formulated (cost as a factor of two or more different weight measures), then the correlation between the technical inputs would be an issue.

The statistical significance of the correlation establishes the strength of the relationship between one parameter and another. In seeking to model program cost as a function of program weight, it is necessary first to establish that a relationship exists between weight and cost growth, and prove that the observed relationship was not likely a mere coincidence. To test for statistical significance, let N be the number of data points for a technical parameter versus performance parameter pair, and tN-2, α be the Student's t statistic for N-2 degrees of freedom and α = 0.05. Solve the following equation for Rc to find the critical value for Pearson's R to be significant at the 95 percent confidence level with sample size N. Eliminate all parameter pairs where the coefficient of correlation is less than the critical value Rc.

$$t_{N-2,\,\alpha} = \frac{R_c}{\sqrt{(1-R_c^2)/(N-2)}}$$

Identify all pairs of technical measures and program performance parameters that meet the significance criteria, and if the direction of the relationship is known, check that the direction of the correlation makes logical sense. Evaluate and document the evidence (if any) of a causal relationship between the parameter pair. An observed correlation between two variables is grounds for considering the possibility

[10] Pearson's R is a measure of the linear relationship between two variables. Values of Pearson's R range from -1 to +1, whereas -1 denotes a perfect negative relationship (i.e., as one variable increases, the second decreases), +1 denotes a perfect positive relationship (i.e., as one variable increases, so does the other) and 0 indicates the absence of a relationship between the variables.

of a causal relationship, but that possibility must be carefully vetted in light of other information about the nature of the two parameters. Only these pairs that are statistically significant and make logical sense should be carried forward in the analysis described in the following sections.

A.3.2 ALI Development Phase 2

Perform detailed correlation and regression analysis on the technical measure versus the program performance measure pairs (identified in PHASE 1). Plot various data sub-sets for various time intervals to achieve a robust fit. Then, quantify the relationship with a mathematical equation/model.

From within the group of parameter pairs that had statistically significant and logical correlation coefficients, the next step is to examine the linear regression[11] models produced by these candidate pairs. Experience would suggest that relationships between a technical measure and a program performance measure tend to change over time and there is no guarantee that the relationship is linear. The following steps will work to verify that the assumptions associated with using simple linear regression models are met, and then create a series of models that will account for shifts in the relationship over time.

Perform simple linear regression ($y = mx + b$) on a single parameter pair from those identified in PHASE 1. Use the technical measure as the y-parameter (response) and the program performance measure as the x-parameter (predictor); this will put the regression equation in the correct format for PHASE 3. At a minimum, be sure to capture the Standard Error, R2, slope and intercept coefficients, and their p-values. It will also be important to view the Normal Probability, Residual, and Line Fit Charts. Note, it has been assumed that the data considered here is Normally Distributed.

Examine the Line Fit Chart to see whether data clusters emerge, or whether the data scatter shows a non-linear relationship between the technical parameter and the program performance parameter (The Normal Probability Chart and the Residual Charts will also show when the linear model is not appropriate for use, however the Line Fit Chart is the most literal representation of the model versus the data.). Relying purely on the R2 could potentially lead to selection of a set of parameters that are not suited for linear regression models. Figure A-6 shows scatter plots for three hypothetical parameter pairs. In each case, the R2 value is 0.50, however only the leftmost scatter plot displays a relationship that is linear. For the other two plots that show a curvature in the data, transformations on the data can be applied to achieve linearity. Logarithmic and Exponential regression models have successfully been used in the past to model these types of relationships. Statistics textbooks may be helpful in providing more detailed methodology for running the linear regressions on transformed data.

[11] Various transformations on the data (logarithmic, natural log, exponential, etc.) can be applied in some cases to produce a better fit. However, based on SEDIC experience using technical measures/TPMs to model overall program performance, simple linear regression models have produced the best results based on principles of parsimony and goodness of fit. By combining independent simple linear regression models, segregated by systems engineering milestones, a logical relationship over time has been achieved that cannot be surpassed using other regression models.

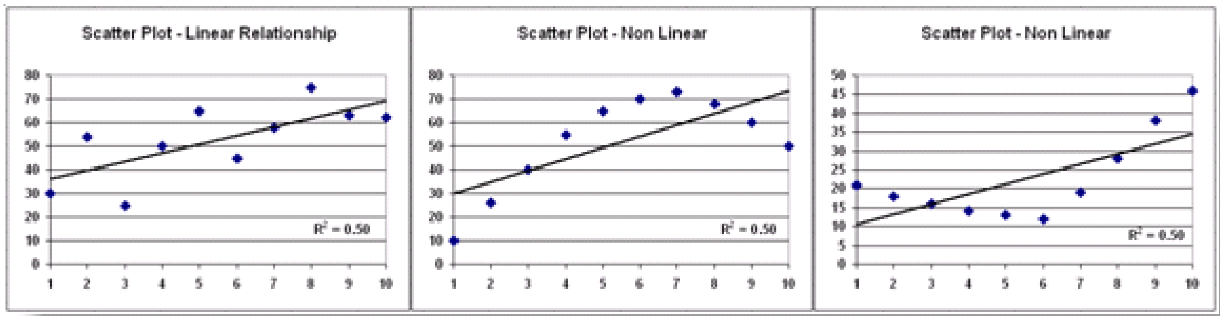


Figure A-6 - Line Fit Chart Examples

To examine shifts in the relationship between a parameter pair over time, group the data by systems engineering milestones [i.e., Contract Award (CA), Preliminary Design Review (PDR), Critical Design Review (CDR), First Flight (FF), Initial Operational Capability (IOC)]. The relationship over time can also be examined by grouping the data by monthly intervals (12-month, 24-month, 48-month, etc.). It may also help to create scatter plots of the technical measure over time and the program performance measure over time to see whether the naturally occurring time intervals present themselves in the data. Once the data has been grouped, re-run the regression on each group to yield several independent linear models.

Check to see whether the regression lines change appreciably between the various time intervals (i.e., does the slope get steeper or shallower between intervals?). Be sure to note the scale of the x and y axes in these charts, as many charting utilities automatically scale the axis to fit the data, thereby potentially obscuring differences between charts.

Review regression coefficient p-values for the slope and intercept – if the p-value is less than .05, then the coefficient is considered statistically significant.

- Check to see that the slope is in a logical direction (if known).
- If the intercept is not significant (0 is in the 95 percent confidence interval and the p-value is greater than .05), re-run the regression forcing the y-intercept = 0.
- Ensure normality and randomness of the residuals, and check that the model fits the data.
- Check that the model is based on a real underlying correlation by taking the square root of R^2 and testing it against the critical value R_c obtained by solving the equation in PHASE 1 for the current data subset.

If all of the above criteria cannot be satisfied, try further segregating the data based on characteristics that may differentiate between programs, taking care to keep enough data in the sub-sets for meaningful analysis to be possible. Examples of categories used for NAVAIR aircraft programs include:

- Mission Type
- Program Executive Office (PEO)
- Conventional Take-Off and Landing (CTOL) vs. Vertical Take-Off and Landing (VTOL)
- Fixed Wing vs. Rotary Wing
- Other categorizations based on naturally occurring groups within the technical measure. For instance, some platforms use a dynamic NTE weight limit that computes this limit as a function of some other characteristic vs. platforms that manage program weight with a static, hard NTE limit.

For a category to be complete, it must span the entire timeline under analysis. Though the models may encompass a subset of the total population, the utility of the ALI Tool lies in its ability to project cost growth, based on a program's current weight status, to any future point in the development cycle (up to IOC).

If all of the above criteria still cannot be satisfied, try segregating the various sub-sets using additional different time intervals, or possibly combine two adjacent time intervals into one interval. Investigate outliers which may be affecting the regression model. If it can be proven that the outlier represents a data anomaly, it may be justifiable to eliminate it from the analysis. However, be aware that in an uncertain and often volatile environment such as aircraft development, many outliers represent real results that if removed, would bias results.

Repeat PHASE 2 for each of the parameter pairs found from PHASE 1. Only the models that meet all of the above statistical criteria are carried forward into the next phases. For the ALI Tool, the only parameter pair to satisfy all constraints was the "percent cost growth" and "percent weight growth from initial weight estimate" parameters. These parameters were initially broken out into the following four time intervals based on systems engineering milestones:

- CA to PDR
- PDR to CDR
- CDR to FF
- FF to the attainment of IOC

Ultimately, the ALI Tool will select the category (i.e., out of all the groups that had a set of models spanning the entire development cycle) that exhibits the highest level of accuracy for the historical data. Model selection by the ALI Tool is further explained in PHASE 5.

A.3.3 ALI Development Phase 3

Use the models from PHASE 2 to generate program performance contours versus time and check for logical relationship trends.

The ALI Tool seeks to project the performance outcome of a project given a known technical measure status such as weight growth to date. In PHASE 3, the regression models developed in PHASE 2 are used to determine what technical measure value would yield a range of performance outcomes. For each of the models that met all statistical criteria from PHASE 2, substitute incremental values for the program performance parameter (x) and solve for y (each model is made up of multiple equations; one for each time grouping).

List the values calculated in a table of technical performance values based on the time interval and incremental program performance value. For example, if percent empty Weight Growth (%WG) is the technical measure and percent %CG is the overall program performance parameter, the following table would be produced:

% CG	Corresponding % WG			
	PDR	CDR	FF	IOC
-25%	-2%	-2%	-2%	0%
0%	0%	-1%	-1%	1%
25%	1%	0%	0%	2%
50%	3%	2%	1%	3%
100%	5%	4%	4%	4%
200%	11%	10%	8%	8%

Figure A-7 - Contour Data Table (in %)

Note in the above example, the time groupings are labeled according to the milestone at the end of the interval (i.e., PDR is from CA to PDR, CDR is from PDR to CDR, FF is from CDR to FF, and IOC is from FF to IOC). With the data contained in this table, it is possible to create a contour chart showing which values of the technical measure (%WG) will yield a certain performance outcome. Plot these values such that the time intervals are arranged on the x-axis, the technical measure values are arranged on the y-axis, and the incremental program performance values are the data series within the chart. An example contour chart is shown in Figure A-8.

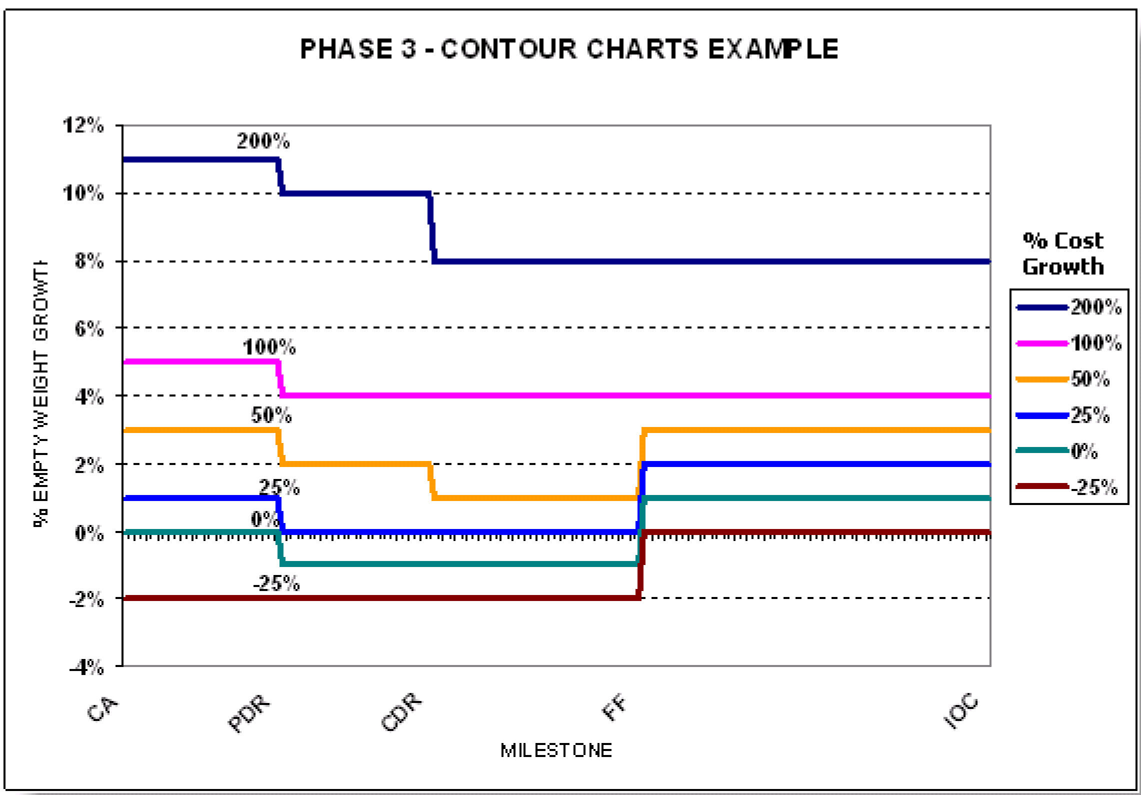


Figure A-8 - Program Performance Contour Plot

Determine which models produce contours that conform to subject matter expert expectations, and meet the statistical and logical requirements from the previous phases. Ideally, the parameter pair for the model should have a plausible causal relationship. Only these models are carried forward into PHASE 4.

A.3.4 ALI Development Phase 4

Perform Verification & Validation (V&V) on the models to confirm they are appropriate for use.

Throughout PHASES 2 and 3, the models have been checked for statistical significance and adherence to logic. Due diligence would necessitate additional tests to ensure each model that will be used in the ALI Tool is reasonably accurate and applicable to the population.

As in PHASE 2, the models all need to have slope coefficients that are significant at the $\alpha = 0.05$ level. If the intercept is included in the model, it too must have a p-value less than 0.05, otherwise there is not sufficient evidence in the data to set the intercept to something other than zero. Confirm also that the models associated with a particular category are significant across all time intervals. If not , there will be a void in the model output that will severely reduce the usefulness of the tool.

To assess the average accuracy for each model, take note of the Standard Error (StdErr). This statistic, which can be found as part of the regression data output from PHASE 2, indicates the average error between the regression's predicted value and the actual data point used in fitting the regression, providing a measure of the model's accuracy. For the weight ALI Tool, there were ultimately three time intervals for each category (the PDR and CDR intervals were eventually combined into a single interval to improve the fit and accuracy of the model). Averaging the StdErr from the three time interval models produced a metric of the overall accuracy of the models within a single category.

The level of detail displayed in the ALI Tool depends on the accuracy of the models. A detailed tool that displays the cost growth contours will need to have a StdErr no more than 20 percent. The detailed tool, due to the nature of the information displayed, can be used to estimate the expected cost savings for a given reduction in weight (see PHASE 5 for a description of the ALI Detailed Tool). This treads into the territory of traditional cost benefit analysis, something that the ALI Tool is supposed to trigger in some cases, but not replace. If the models have an average StdErr between 20 percent and 50 percent, then a "Tripwire" version of the tool can be developed. The ALI Tripwire Tool takes away the specificity of the contours, providing only a range of cost growth values associated with a given percent weight growth. Rather than individual contours, the Tripwire Tool has regions marked by green, yellow, and red to indicate cost growth ranges that are acceptable, borderline, or unacceptable to the user community (see PHASE 6 for a description of the ALI Tripwire Tool). If the StdErr is greater than 50 percent, then the model is not accurate enough for either application. In this case, the methodology must be repeated with a new set of parameters.

The regression models for a particular categorization should also be reasonably robust, and should be tested to ensure there is not a single program that has undue influence on the slope or intercept of the regression line. The Jack Knife method to sensitivity testing assesses the change to the regression fit upon removal of a single point from the data behind the original regression fit. In the case of the ALI Tool, rather than removing a single point, it is done by removing all the weight data associated with a single particular program and re-calculating the regression equations[12]. This modified Jack Knife procedure has to be performed on the models for each category and time interval.

[12] Performing sensitivity analysis on a single data point would not yield meaningful results in this case. Most statistical software packages allow the user to compute the Cook's Distance, or Cook's D, for each data point, which is a measure of a point's "pull" on the regression line. If Cook's D exceeds a certain threshold, it signifies that the regression line is heavily influenced by that particular point. In the case of a regression with hundreds of data points, Cook's D for each point becomes exceedingly small, and nowhere near the critical value. In the case of the ALI Tool, the concern is moreover whether an entire program, which has several data points, is having an undue influence, rather than any particular data point.

Due to the removal of full program data sets rather than single data points, there are no statistics and critical values to compare for the sensitivity test. The easiest way to assess the results is to plot the new regression equations and compare them to the original regression equation. All regression equations should remain relatively constant: slope and intercepts may change slightly, but they should be relatively close to the original equation, which is shown in the solid black line in the following example plots. In

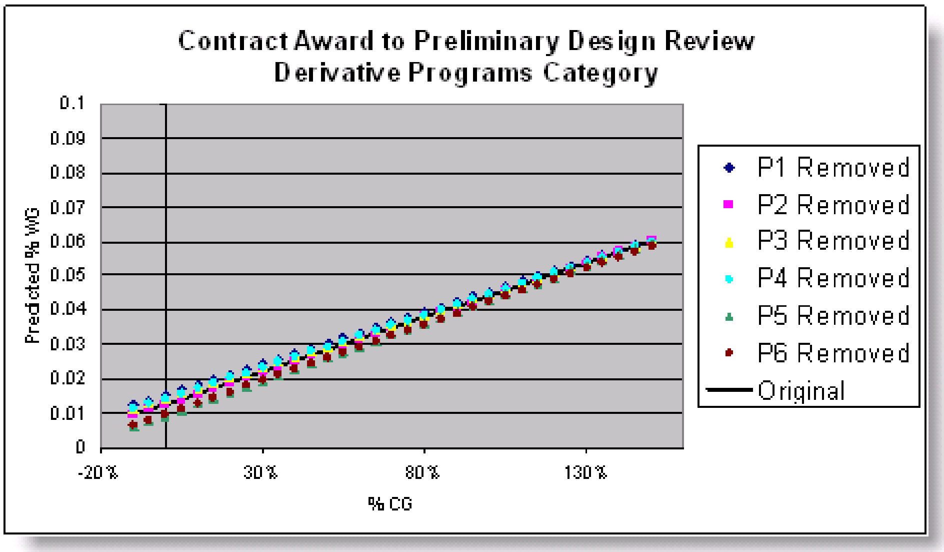


Figure A-9, the Derivative model shows a nearly ideal case.

Figure A-9 - Ideal Jack Knife Sensitivity Results

If any equations stand out from the other regression equation lines, or if the slope changes sign, then the model fails. In the following case, the VTOL model fails because one program has an overwhelming influence on the slope of the original regression model. Notice in Figure A-10, how with the removal of P6 results in a dramatic departure from the original regression model and the sign of the slope changes. Because the removal of the other programs (P1-P5) did not result in significant changes in the regression, P6 appears to be the dominating the overall fit. Some analysis can be done to determine whether P6 is representative of "typical" VTOL programs. If P6 is an atypical VTOL program, tool developers may consider re-fitting the regression without the one outlier.

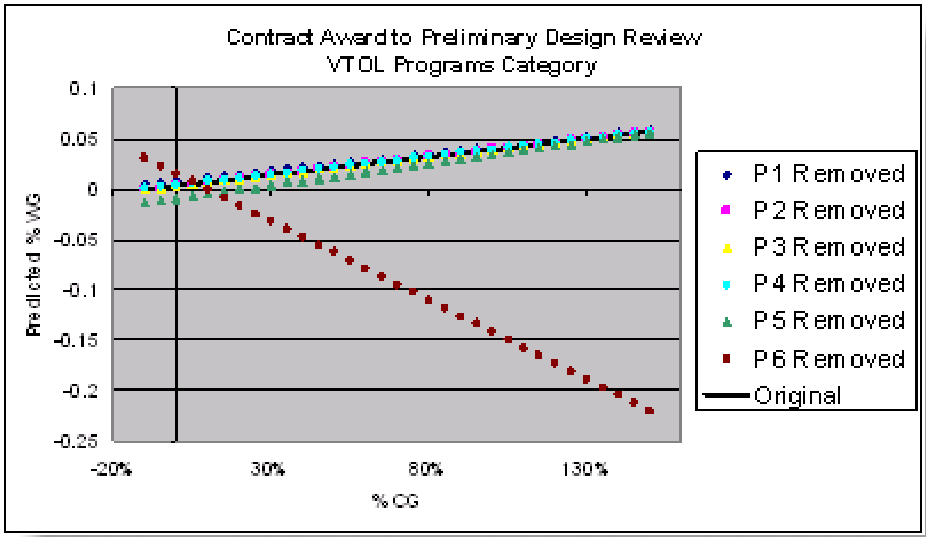


Figure A-10 - Poor Jack Knife Sensitivity Results

It should be noted that the weight versus cost ALI Tool developed by the SEDIC has been unable to achieve the 20 percent accuracy level (to date). Therefore, the ALI Tripwire Tool is the only tool currently being used at NAVAIR. The SEDIC is in the process of trying to collect additional data in an attempt to achieve an accuracy level that would permit the use of the ALI Detailed Tool.

A.3.5 ALI Development Phase 5

Combine contours passing PHASE 4 with existing program technical measure charts to produce the ALI Detailed Tool.

While it was appropriate to transform the technical data into dimensionless values for the purposes of the analysis, dimensionless measures may not be the typical format users are familiar with. Recall that the table and contour chart produced in PHASE 3 (see Figures A-7 and A-8) showed the dimensionless measures (e.g., %WG) on the y-axis. To reduce the difficulty of understanding and correctly interpreting the tool's outputs for the user, PHASE 5 converts the dimensionless values needed for the regression models into discrete, quantitative values according to the user's specific program. In effect, the y-axis is translated back into the units the user is most familiar with (i.e., those that are applicable to their specific program).

As an example, consider the case with program weight data—most programs track this technical measure in pounds. Specifications and Key Performance Parameters (KPPs) also listed this technical measure in pounds; therefore, it made sense for the tool output to match that convention. Using the following equation and program-specific initial weight data, percent weight growth is converted into an estimate of the weight of the aircraft associated with a series of cost outcomes.

Weight = Orig. Wt. Est. + (%WG X Orig. Wt. Est.)

Returning to the contour table from PHASE 3, this transformation will produce the table in Figure A-11.

%CG	Corresponding Weight (lbs)			
	PDR	CDR	FF	IOC
-25%	29159	28984	29003	29800
0%	29577	29377	29342	30035
25%	29994	29770	29682	30270
50%	30412	30163	30021	30505
100%	31247	30949	30701	30974
200%	32918	32522	32059	31914

Figure A-11 - Contour Data Table (in lbs.)

Now that the program performance contours developed in PHASE 3 are expressed in pounds, these contours can be overlaid onto the program's existing technical measure chart to produce the "ALI Detailed Tool". This tool can be used to plot the status of the program's existing technical measure and determine the corresponding level of historical program performance expected as a function of time. This Detailed Tool, however, must pass a more stringent level of statistical significance and accuracy tests as addressed in the previous phase.

The output from this ALI Detailed Tool (see Figure A-12) can be used to determine the approximate overall program performance that can be expected as a result of improvements or changes to the technical measure. This information can be used to perform cost-benefit analysis on change initiatives.

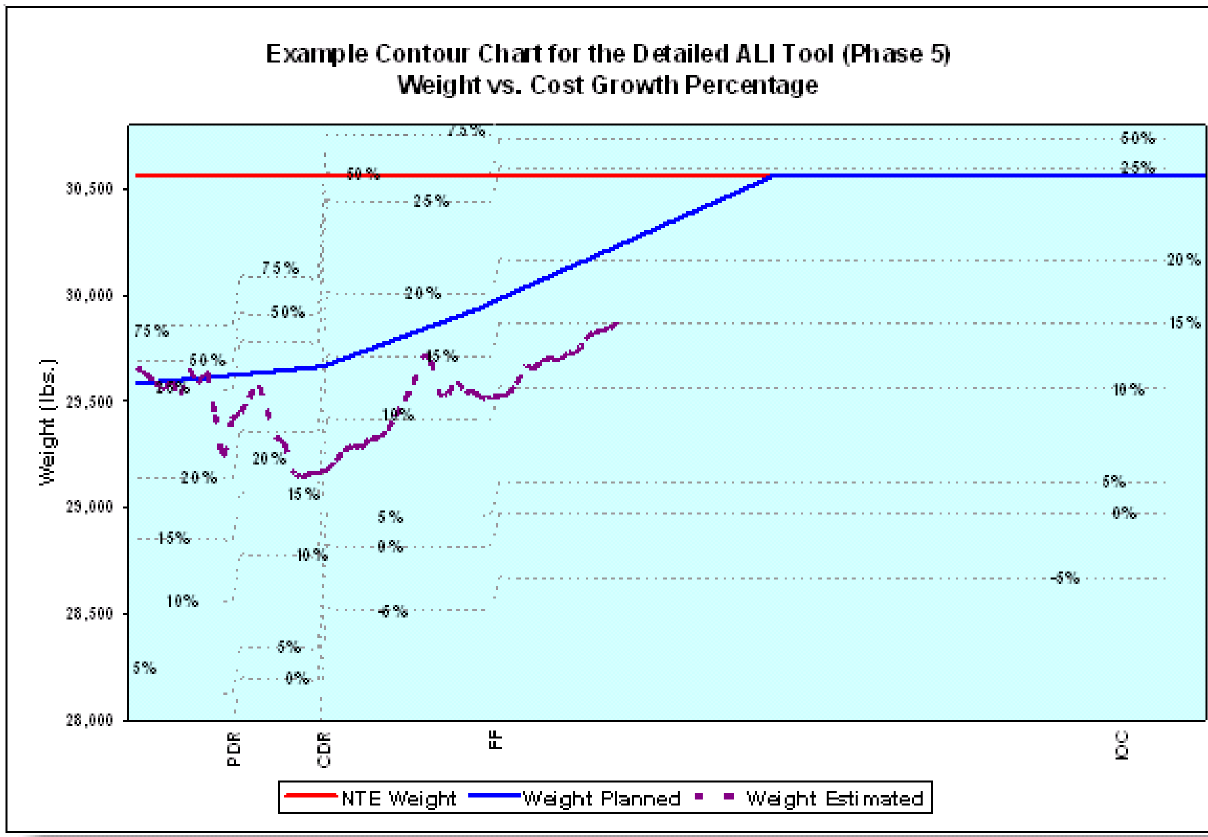


Figure A-12 - ALI Detailed Tool Output

In Figure A-12, a program's NTE weight limit (in red), weight plan (in blue), and estimated weight (dotted purple line) are plotted against several different cost growth percent contours. According to this example, the program's current estimated weight is about 29,850 pounds, which corresponds with a cost growth of roughly 15 percent. Users can plot future projections of weight growth from this chart based on the latest technical aspects of their program to see how future perturbations in weight growth (i.e., unforeseen weight growth notices from the contractor or future weight growth mitigation efforts) will impact overall program cost growth. This will help engineers and program managers to determine the best potential courses of action related to this technical aspect of the program. Using multiple ALI Detailed Tools together (i.e., weight growth, staffing performance, requirements volatility, SLOC growth, etc.) will help engineers and program managers identify the technical aspects of their program that are causing the most negative impact on overall program performance. This will allow them to focus program resources where they are needed most.

A.3.6 ALI Development Phase 6

Incorporate final models into ALI Tripwire Tool.

Using the accuracy results from PHASE 4, establish a hierarchy of the models that have passed the V&V steps. The ALI Tool uses only one category's set of regression models over the entire time period covered by the analysis. The model is selected based on inputs from the user, which are exploited to identify which categories apply to the program being evaluated. From the hierarchy, the applicable model with the best accuracy (smallest StdErr) is used to compute the predicted performance outcomes and contours for the ALI Tool.

With the model selected and the appropriate time interval determined for the weight input, the predicted performance outcome can be computed and plotted. In the following example charts, this predicted outcome point is labeled and color coded. The color coding is based on the R^2 value for the specific model used to predict the performance outcome. For the ALI Tripwire Tool, the color coding should be interpreted as follows:

- Light Blue = R^2 < 25% = there is some uncertainty surrounding the placement of the weight status point in relation to the colored regions
- Royal Blue = 25% > R^2 > 50% = the placement of the weight status point in relation to the colored regions is of moderate precision
- Dark Blue = R^2 > 50% = there is high confidence that the placement of the weight status point in relation to the colored regions is accurate

Because the accuracy requirements were not sufficient (StdErr>20%) to support a detailed tool that produced a specific predicted value for cost growth (the performance outcome), the tool from PHASE 5 is adjusted to provide general Red/Yellow/Green ratings rather than the incremental contours as shown in Figure A-12. The thresholds for these regions should be based on user/organization desired program performance levels.

For NAVAIR programs, the Yellow to Green limit was established based on the cost growth percentage that would trigger a minor Nunn-McCurdy breach[13]. The Yellow to Red limit was established based on the current average cost growth for development programs across the organization. This limit was chosen because one of the goals of implementing this tool is to help reduce the average cost growth across all programs in the organization. Using these limits, a Green score indicates the program can expect to execute without requiring a Nunn-McCurdy breach. A Red score indicates the program under evaluation is expected to experience above NAVAIR average cost growth. A Yellow score indicates the program will likely execute better than the NAVAIR average but will still likely have a Nunn-McCurdy breach. This proactive indication of overall program performance will hopefully encourage some preventative action, when required, to arrest a negative trend.

Having these thresholds established and approved by the user community, the contour charts developed in PHASE 5 essentially are reduced to charts with two contours: the Yellow to Green cost growth contour and the Red to Yellow contour. The regions defined by these two contour lines should then be shaded Red, Yellow, and Green as seen in Figure A-13.

---

[13] U.S. legislation known as the Nunn-McCurdy Amendment to the Defense Authorization Act of 1982 mandates that Defense-related procurement programs where the cost of each individual item being purchased reaches 115% of the original contract amount must notify Congress of the overrun. In addition, any program which reaches a unit cost overrun of 25% will be cancelled unless the Secretary of Defense provides convincing and detailed justification to Congress that the program is critical to national security.

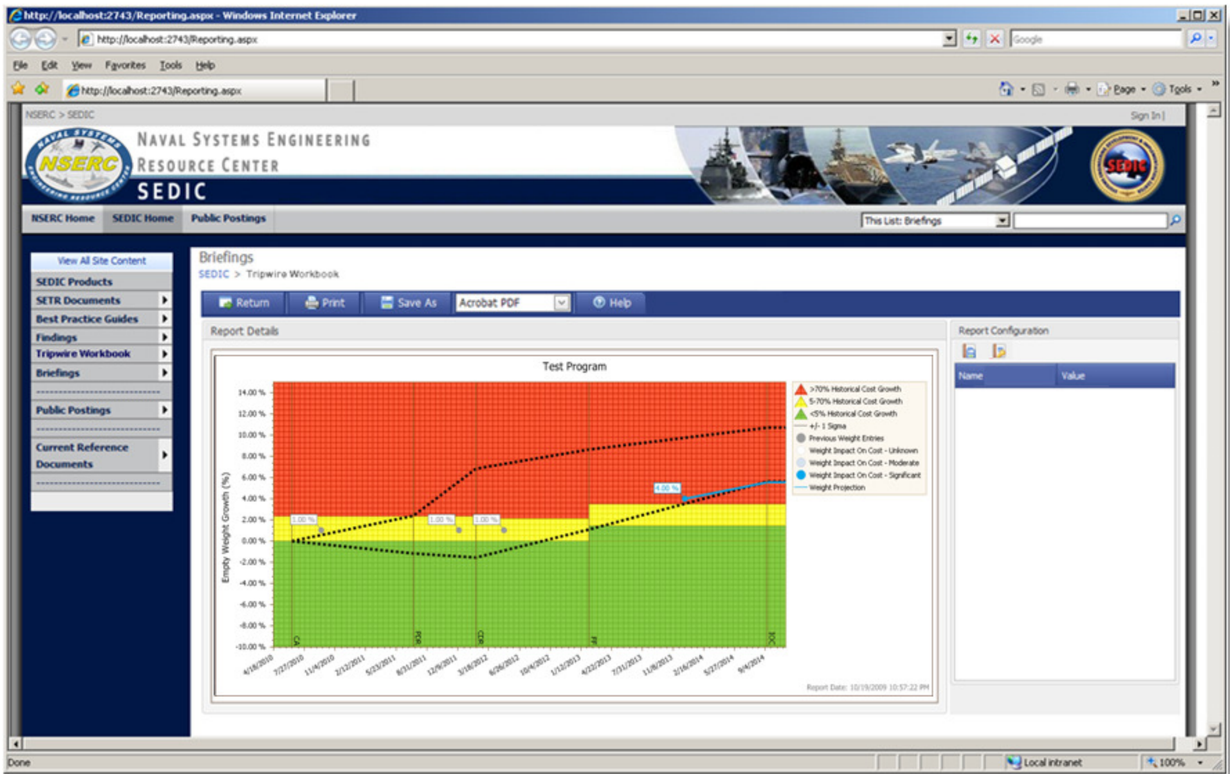


Figure A-13 - ALI Tripwire Tool – Red/Yellow/Green Display

To assist users in projecting future states of the program, and to estimate cost growth at the end of the development cycle, the tool can be programmed to show a projection of the weight growth. The projection lines will be based on historical weight growth trends for programs in the sample data set. Using the sample data set, compute the average and the standard deviation historical rate of change of the technical measure.

The upper and lower range is computed as one standard deviation up and one standard deviation down from the mean, and is shown in the chart as dotted lines (see Figure A-14). This represents a likely range of empty weight status for the program, starting from 0% growth at CA and assuming the program follows historical trends to IOC. This +/- 1-Sigma range accounts for ~70% of the sample population. In other words only 30% of historical programs were able to exhibit weight growth outside of this range.

The weight projection line, shown in Figure A-14 as a solid black line, is drawn to show what the program's empty weight would be in the future if it follows the average historical growth rate. The weight projection line does not take into account how the program got to its current status weight; it applies the historically-observed average growth trend to the current status going forward. The projection line can be used to estimate %WG values at each future time interval and allow the user to determine the corresponding level of program performance (cost growth) associated with the projected %WG. The location of the weight projection line relative to the dotted upper and lower ranges gives the user an idea of how closely their program's technical measure is tracking to historical trends.

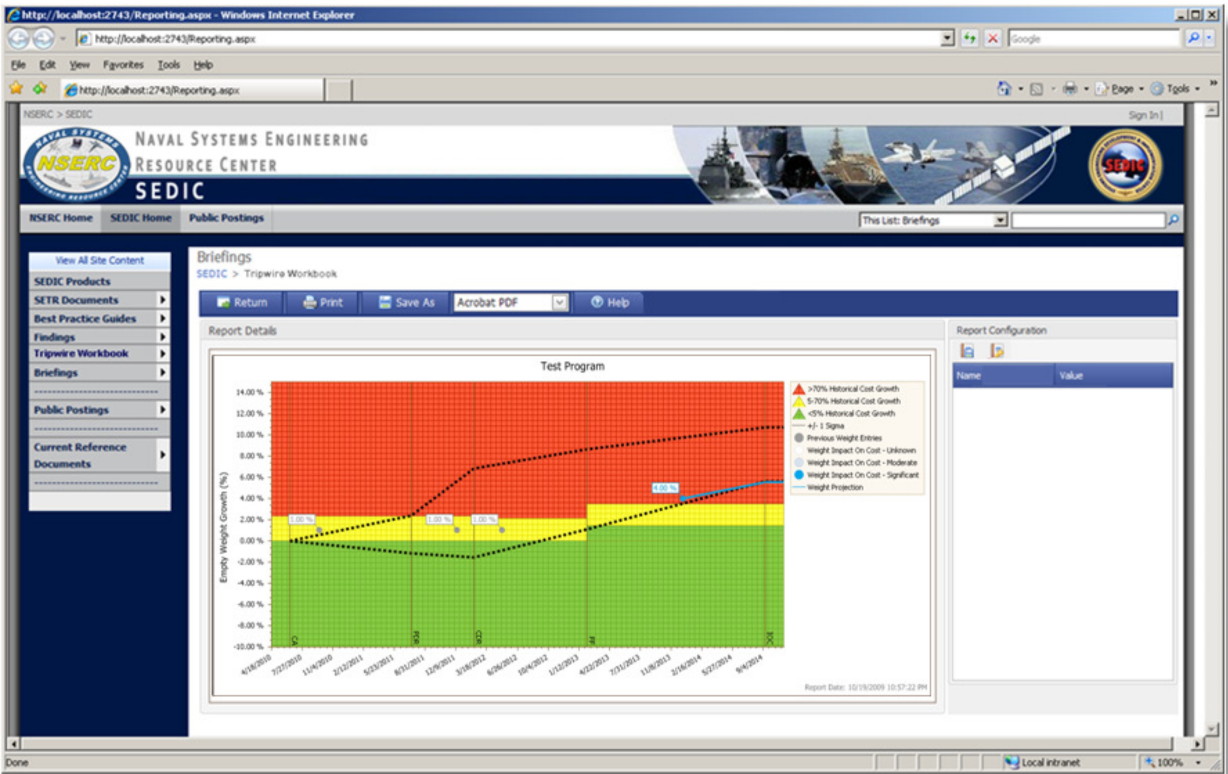


Figure A-14 - ALI Tripwire Tool – Weight Projection Lines

Programs can use this ALI Tripwire Tool to compare their technical measure performance to historical programs and determine the corresponding level of overall program cost growth that can be expected in the future. Coupled with program-specific insight, program managers can use this tool to determine when to initiate investigative action to determine whether corrective action would be cost effective.

A.4 LIMITATIONS OF THE TOOL

These ALI Tools are based on actual demonstrated program performance for a sample of historical programs. As such, the tool will only be applicable to programs that are represented by the chosen sample. For example, at NAVAIR the sample data set consisted of only ACAT I & II aircraft development programs; sub-system upgrades, ACAT III & IV, etc. types of programs likely cannot use this version of the tool.

In addition, overall program performance (cost/schedule/technical) is influenced by a number of different factors. Since a single ALI Tool only accounts for the effect of one technical measure on program performance, the accuracy and utility is limited. Only using a suite of multiple ALI Tools together will provide a more complete understanding of the expected performance of your program – and all the factors that are contributing to that performance (i.e., weight, requirements volatility, staffing, SLOC).

These ALI Tools are intended to be used as a guide by program leadership to proactively initiate investigative action. They are meant to provide an early indication of the magnitude of expected overall program performance and the factors that may be contributing the most. This is meant to assist/direct investigative actions by the program to determine what specific corrective action should be taken to ensure successful program execution. Detailed, program-specific knowledge and insight must be used along with the output from these ALI Tools in order to determine the appropriate action.

A.5 APPLICATION OF THE ALI TOOL WITHIN A PROGRAM

Programs typically receive technical measure/TPM charts on a periodic (often monthly or weekly) basis which provide a status of each technical measure as it relates to achieving the technical requirements set by the program. These TPM charts do not describe how each technical measure contributes towards overall program performance (i.e., % cost growth or schedule growth). The ALI Tool provides an additional gauge of the technical measure from an overall program performance perspective. Programs will now have two different perspectives to measure technical parameters.

The programs will compare the status of each technical measure from a purely technical requirements standpoint (using existing TPM charts) to the status of each technical measure from an overall program performance standpoint (using the ALI Tool). The program will use the ALI Tool to determine the approximate level of program performance (i.e., cost growth) expected at the end of the program. The program can then use the ALI Tripwire Tool to run "future-case" scenarios to see what future technical measure performance will achieve the desired result in terms of final program performance (i.e., cost growth). Using all of this information, the programs will then be able to make informed decisions regarding what areas of their program are contributing the most undesirable effects to overall program performance. They can then initiate investigative actions to determine how best to address these areas and possibly the magnitude of resources that should be applied.

The ALI Tools developed by the SEDIC for NAVAIR are proprietary and are not included as part of this Appendix. However, organizations interested in developing their own ALI Tools can contact the SEDIC for additional information regarding the tool design, user interface, etc. currently being deployed at NAVAIR.

A.6 LOOKING FORWARD

The SEDIC is currently in the process of piloting ALI Tools on various programs at NAVAIR. The data collected as a result of these pilot programs will be used to further refine the methodology and tools outlined in this Appendix. Future revisions of this Appendix are planned to refine this preliminary methodology for developing these ALI Tools.

In addition, the SEDIC is also in the process of developing a method to combine multiple technical measures to form a more robust ALI Tool. Based on models that can account for the interactions between various technical measures, the next generation of ALI Tools will consist of a suite of ALIs that users can easily navigate to see what specific technical areas of their program require the most attention/mitigation in relation to one another.

References:

[A1] Aircraft Airframe Cost Estimating Relationships: All Mission Types; R.W Hess, H.P. Romanoff; http://www.rand.org/pubs/notes/2005/N2283.1.pdf; 1987

[A2] Parametric Equations for Estimating Aircraft Airframe Costs; Joseph P. Large, Hugh G. Campbell, D. Cates; http://www.rand.org/pubs/reports/2006/R1693-1.pdf; 2007

A.7 ACRONYMS

%BNTE	Percent Below NTE Weight
%BP	Percent Below Weight Plan
%CG	Percent Cost Growth
%CWG	Percent Cumulative Weight Growth
%WG	Percent Weight Growth
ACAT	Acquisition Category
ALI	Applied Leading Indicator
CA	Contract Award
CDR	Critical Design Review
CPI	Cost Performance Index
CTOL	Conventional Take-Off and Landing
EAC	Estimate At Complete
EAC ML	Estimate At Complete (Most Likely)
FF	First Flight
IOC	Initial Operational Capability
KPP	Key Performance Parameter
LI	Leading Indicator
NAVAIR	Naval Air Systems Command
NTE	Not-to-Exceed
OCBB	Original Contractor Budget Baseline
OPEVAL	Operational Evaluation
PDR	Preliminary Design Review
PEO	Program Executive Office
SEDIC	Systems Engineering Development and Implementation Center
SLOC	Software Lines of Code
SPI	Schedule Performance Index
StdErr	Standard Error
TPM	Technical Performance Measure
UAS	Unmanned Air Systems
V&V	Verification and Validation
VAC	Variance at Complete
VTOL	Vertical Take-Off and Landing

APPENDIX B – Human Systems Integration Considerations

This appendix describes ongoing research investigating leading indicators for human systems integration, within the overall systems engineering practice. The research is being performed by the MIT Systems Engineering Advancement Research Initiative (SEAri), and is sponsored by the US Air Force Human Systems Integration Office (AFHSIO). One goal of the research will be to augment this guide, and the preliminary work is described below. The results of the research will provide input to the next version of this guide, and may lead to updates of the existing measurement specifications, with possible addition of one or more new indicators.

Human Systems Integration (HSI) is the integrated, comprehensive analysis, design and assessment of requirements, concepts, and resources for system Manpower, Personnel, Training, Environment, Safety, Occupational Heath, Habitability, Survivability and Human Factors Engineering. HSI considerations are an integral part of the overall systems engineering process in all engineering application sectors, and are critically important for defense systems and other large scale systems where humans are involved and impacted.

HSI is tightly coupled with the systems engineering process, particularly in large defense and government programs, making it challenging to determine if HSI is sufficiently considered to ensure a successful program. By its nature HSI must be considered in early phases of acquisition and development, and given adequate focus throughout development, fielding, and operations. The complexity of military systems and systems-of-systems (SoS) motivates this need for "a robust process by which to design and develop systems that effectively and affordably integrate human capabilities and limitations" [B1]. Additionally, adequate management of human considerations within the system contributes to increasing total systems performance and decreasing total ownership costs by reducing human errors, optimizing interface design, and eliminating occupational hazards.

A challenge exists for program leadership to predicatively assess whether HSI considerations have been adequately addressed to result in overall systems effectiveness in regard to the domains of HSI. The current set of systems engineering leading indicators has weak characterization with regards to human systems integration; new empirical research is examining how these can be modified and extended to more effectively address HSI considerations. For example, the tracking of requirements changes—which already occurs in the quantification of leading indicators—can be extended to specifically track the subset of HSI requirements.

HSI incorporates functional areas, referred to as domains. The US Air Force defines HSI Domains as: Manpower, Personnel, Training (sometimes combined into MPT), Human Factors Engineering, Environment, Safety, and Occupational Health (the previous three are commonly grouped as ESOH), Survivability, and Habitability. Other organizations use similar areas in defining HSI activities. A brief description of these is shown in Table B-1 below.

Table B-1 – HSI Domain Descriptions [B2]

Domains	***Description***
Manpower	The number and mix of personnel (military, contractor) authorized and available to train, operate, maintain, and support each system acquisition.
Personnel	The human aptitudes, skills, knowledge, experience levels, and abilities required to operate, maintain and support the system at the time it is fielded and throughout its life cycle.
Training	The instruction and resources required to provide personnel with requisite knowledge, skills, and abilities to support the system.
Human Factors Engineering	The comprehensive integration of human capabilities and limitations (cognitive, physical, sensory, and team dynamic) into system design, development, modification and evaluation to optimize human-machine performance for both operation and maintenance of a system. Human Factors Engineering designs systems that require minimal manpower, provide effective training, can be operated and maintained by users, and are suitable and survivable.
Environment	Environmental factors concern water, air, and land and the interrelationships which exist among and between water, air, and land and all living things.
Safety	Safety factors are design and operational characteristics that minimize the possibilities for accidents or mishaps to operators which threaten the survival of the system.
Occupational Health	Occupational Health factors are design features that minimize risk of injury, acute and/or chronic illness, or disability, and/or reduced job performance of personnel who operate, maintain, or support the system.
Survivability	The characteristics of a system that reduce risk of fratricide, detection, and the probability of being attacked; and that enable the crew to withstand man-made or natural hostile environments without aborting the mission or suffering acute and/or chronic illness, disability, or death.
Habitability	Factors of living and working conditions that are necessary to sustain the morale, safety, health, and comfort of the user population which contribute directly to personnel effectiveness and mission accomplishment, and often preclude recruitment and retention problems.

The MIT SEAri research team is exploring how to best augment the current measurement specifications, through empirical research involving interviews with experts and case studies of programs. As a first step, the domains are being investigated with regard to what types of insight would be important.

Examples of preliminary ideas are:

Manpower: Early identification of an escalation or reduction in manpower requirements may warn of an unexpected increase in system complexity. Initial insights provided by a manpower assessment of the entire system life cycle may bring to light hidden costs often overlooked in the maintenance and disposal phases.

Training: Initial insight into the adequacy of training resources may inform managerial decisions and improvements, resulting in an increased speed of adoption among the user population, the reduction of user errors, or alternatively the reduction in unnecessary training costs.

Environment: A holistic evaluation of the environment within which a system operates, the environment's possible states of flux, and system impacts on that environment provide valuable insight that will influence the strategic design and operation of that system. For example, recognition of probable material resource scarcity or the identification of generated noise pollution, may inform redesign of major components.

Figure B-1 shows a possible approach where a field is added to the specification to include HSI considerations and other information is added to fields such as the information category. The concept in the figure is notional; the research team is currently investigating possible options with a goal of integrating as much HSI information as possible, such that it is not necessary to define many new leading indicators specifically for HSI

Requirements Validation Rate Trends	
Information Need Description	
Information Need	Understand whether requirements are being validated with the applicable stakeholders at each level of the system development.
Information Category	1. Product size and stability – Functional Size and Stability 2. Also may relate to Product Quality a performance (relative to effectivenes efficiency of validation)
Measurable Concept and Leading Ins	
Measurable Concept	The rate and progress of requirements va
Leading Insight Provided	Provides early insight into level of underst customer/user needs: • Indicates risk to system definition d inadequate understanding of the cus needs • Indicates risk of schedule/cost overr delivery changes, or user dissatisfac

Requirements Validation Rate Trends	
Information Need Description	
HSI Considerations	***Validate with the stakeholders that, across all system elements, requirements provide significant coverage for relevant HSI domains.***
Information Need	Understand whether requirements are being validated with the applicable stakeholders at each level of the system development.
Information Category	1. Product size and stability – Functional Size and Stability 2. Also may relate to Product Quality and Process performance (relative to effectiveness and efficiency of validation) ***3. Product success relative to applicable HSI domains***
Measurable Concept and Leading Insight	
Measurable Concept	The rate and progress of requirements validation.
Leading Insight Provided	Provides early insight into level of understanding of customer/user needs: • Indicates risk to system definition due to inadequate understanding of the customer/user needs • Indicates risk of schedule/cost overruns, post delivery changes, or user dissatisfaction

Figure B-1 – Example of adapting leading indicator specification to include HSI considerations.

Once the leading indicators have been enhanced for HSI considerations, it may also be important to define next level measurement detail to be looked at through further data analysis, and graphical portrayal of the information. A notional example is shown in Figure B-2, where requirements validation information might be examined more closely for requirements that are allocated to the HSI domains. Graphical constructs can provide assistance in seeing where variances against historical data in similar systems occur.

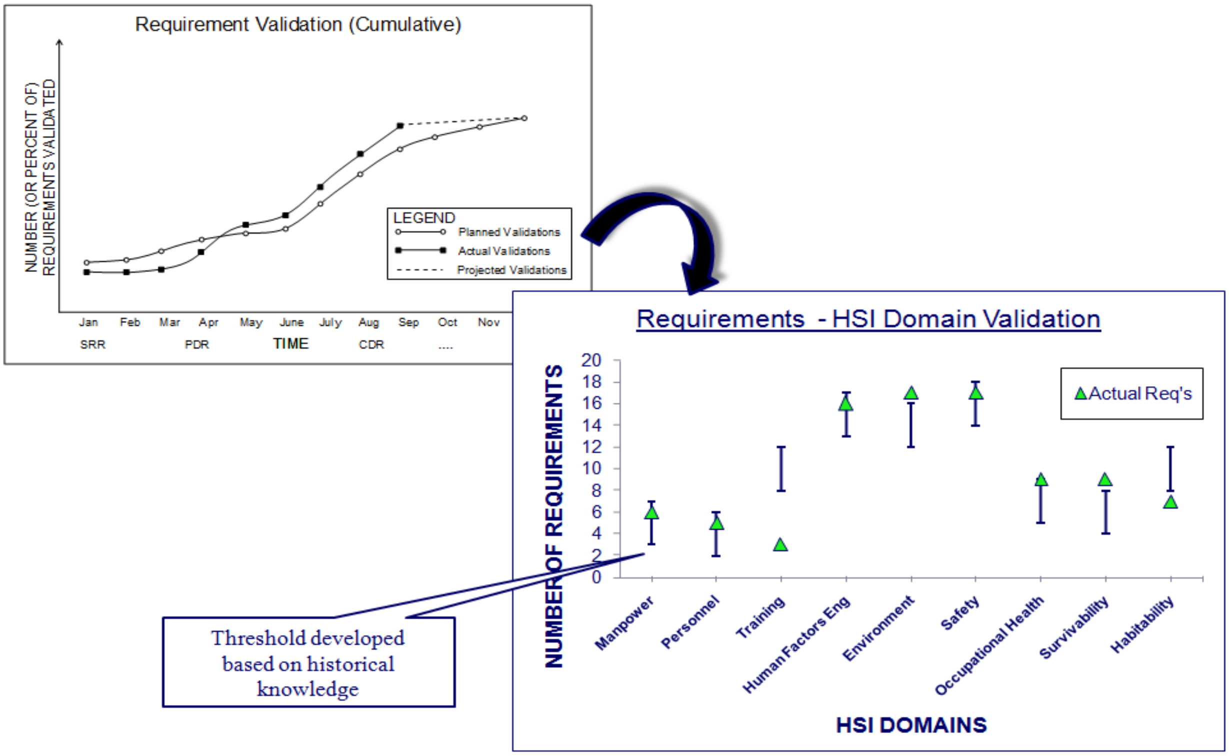


Figure B-2 – Graphical aid to assess number of HSI requirements validated against accepted historical range for similar programs.

The research has also identified the importance of ***Soft Indicators*** in assessing the effectiveness of HIS as part of the systems engineering effort. A soft indicator is defined as a piece of qualitative, difficult-to-measure information, whose existence indicates early-on program success or failure. For example, the existence and adequacy of a clearly identified HSI effort within the Systems Engineering Plan, including strategies related to the development of HSI specifications, is an early indicator of success that is difficult to describe on a quantifiable scale. These types of qualitative indicators provide management with a useful tool when dealing with projects that exist in complex situations. The identification and use of soft indicators increases the portfolio of tools available to management for predictive performance of HSI.

The early research effort has identified six areas which are likely to have relevancy for augmenting the set of current systems engineering leading indicators, including use and interpretation guidance for these indicators. In examining the soft indicators, the team may discover the need to add one or more additional leading indicators to the guide. As of publication of the guide, it is expected that the continued empirical research may identify additional areas of importance. The six soft indicators are:

1. ***Allocation of Requirements to HSI Domains.*** During the definition of the system and subsystem level requirements it is very important that performance and behavioral requirements covering all of the human systems integration domains be adequately specified. Where requirements relate to complex interdependencies between the human and the system, potential exists for using the requirements-related leading indicator information to monitor negative trends in requirements growth, volatility, verification and validation specific to this focus. Indicators of high volatility in the allocation of requirements could warrant further investigation to determine if appropriate HSI personnel are involved since requirements that are misallocated to humans (or not to humans when they should be) in the early phase can result in major issues downstream.

2. ***Impact Assessment of Allocations/Allocation Changes***. A related observation is that highly effective organizations have a mature process for impact assessment of requirements allocation (usually once some level of requirements stability has been established). Of particular interest will be assessment of changes related to allocation from hardware/software to human, or human to system. Requirements volatility, verification and validation trend indicators will be of interest, with lower level details to support investigation.

3. ***Adequacy of Stakeholder Involvement.*** Within the overall footprint of HSI there are many types of stakeholders that need to have a voice in the system development. Leading indicators related to staffing and work product approval, for example, may show whether all necessary stakeholders are (or are not) in key activities such as participating in requirements review and design reviews. The absence of HSI personnel, particularly in early system definition activities, will likely result in inadequate consideration of the HSI requirements, possibly leading to significant system failures.

4. ***Orientation for HSI in Engineering Organization.*** Another soft indicator that has been highlighted by systems engineering experts as important is the existence of an orientation within the engineering organization for the HSI perspective. Whether or not there are dedicated HSI specialists, there is variation in engineering organizations in regard to how much focus and priority is placed on the HSI considerations. A highly effective engineering team has cognizance of the full set of HSI domains, and the organization (customer and contractor) proactively establishes HSI related technical measurements (KPP, MOE, etc.), for example human survivability MOEs. Within the mature engineering organization there will be individuals with roles and authorities to provide coverage for all of the domains appropriate to each program.

5. ***Adequacy of Domain-specific Expertise.*** HSI includes multiple domains which require unique expertise that is rarely found in one individual. A soft indicator of a less mature organization will be assignment of all the domains to one specialty unit or specialty engineer. Leading indicators that look at staffing in regard to coverage of the HSI domains can provide insight to whether the systems will ultimately perform well as related to the domains. Failure to provide coverage for these necessary skills and roles will likely have negative implications for performance in the operational system.

6. ***Understanding Situational Factors Impacting HSI.*** In the definition and specification of leading indicators, it is important to understand underlying situational factors that may impact the effectiveness of HSI and accordingly how the leading indicator information may be interpreted. Factors may include severity of threat or operational environment, complexity of system interfaces, newness/precedence of system technologies, and socio-political relationship of constituents in SoS. As an example, where the system has very complex system interfaces involving new technology and inexperienced users, the *interface trends* leading indicator may need to be reported and monitored on a more frequent basis, and interface allocation to humans designed to accommodate the inexperience user base.

References:

[B1] U.S Air Force (2008). "Air Force Human Systems Integration Handbook", Air Force 711 Human Performance Wing, Directorate of Human Performance Integration, Human Performance Optimization Division, Washington, DC: Government Printing Office. (Draft)

[B2] Rhodes, D.H., Ross, A.M., Gerst, K.J., and Valerdi, R., "Leading Indicators for Human Systems Integration Effectiveness," 7th Conference on Systems Engineering Research, Loughborough University, UK, April 2009, http://seari.mit.edu

APPENDIX C – Early Identification of Systems Engineering Related Program Risks

The Systems Engineering Research Center - University Advanced Research Consortium (SERC_UARC) has recognized a number of critical success factors related to Systems Engineering program risks. The goal of the effort is to identify of SE-Related Program Risks (SERPR) and codify them in tools.

These tools are "SE Performance Risk Tool" (SEPRT) and the "SE Capability Risk Tool" (SECRT). They are intended to provide DoD program managers with early warning of any risks to achieving effective Systems Engineering, especially at major milestones. Users of the tools would be Systems Engineering organizations at the request of DoD program managers or the contractor's management in pro-active rather than re-active sense for planning.

There are three important points that need to be made about these risks.

- The risks are generally not indicators of "bad Systems Engineering." Although SE can be done badly, more often the risks are consequences of inadequate program funding (Systems Engineering is often the first victim of an under-budgeted program), of misguided contract provisions (when a program manager is faced with the choice between allocating limited Systems Engineering resources toward producing contract-incentivized functional specifications vs. addressing key performance parameter risks, the path of least resistance is to obey the contract), or of management temptations to show early progress on the easy parts while deferring the hard parts till later.
- Analyses have shown that unaddressed risk generally leads to serious budget and schedule overruns.
- Risks are not necessarily bad. If an early capability is needed, and the risky solution has been shown to be superior to the alternatives, accepting and focusing on mitigating the risk is generally better than waiting for a better alternative to show up.

The operational concept for the SERPR tools includes the following primary responsibilities, authority, accountability (RAA):

- Primary assessment consumers: Persons with management responsibility for program results, i.e., Contractor PM, DoD PM/PEO, oversight personnel
- Primary assessment conveners, monitors: Chief Engineers, Chief Systems Engineers, respectively.
- Primary assessors: Independent experts

The Figure C-1 below shows the SERPR tools operational concept (for each stage of development) in a work flow sense. Assuming a formal risk assessment effort is judged to be worth the effort, the formal program plans are developed. The program and the associated plans are evaluated versus critical success factors codified in the SERPR tools.

Following a successful evaluation using the SERPR tools, the figure depicts setting the appropriate Leading Indicators control limits for the program based on corporate experience, prior to proceeding with the program execution. The program execution is continuously evaluated, and so long as it is within the allocated control limits, it proceeds without corrective action.

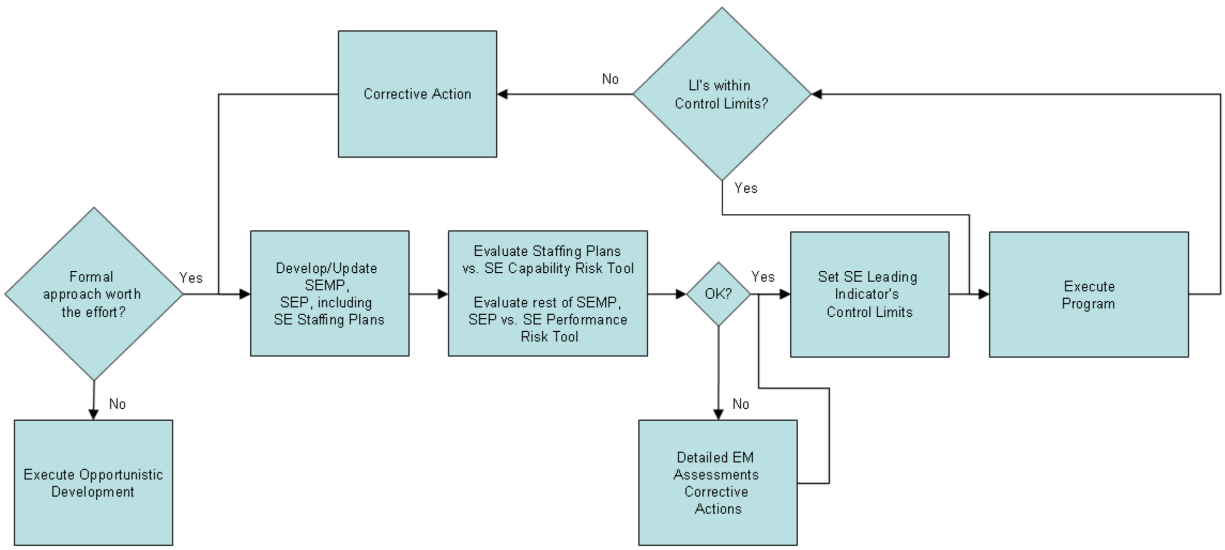


Figure C-1 – Operational Concept for SERPR Tools

The critical success factors considered by the SERPR tools are organized under higher level goals as shown in Table C1. The critical success factors reflected in the SERPR tools have been compared to the Leading Indicators of this guide in Table C2.

The Systems Engineering Leading Indicators and SERPR tools, while related, are somewhat orthogonal to each other; the SERPR tools are designed to be evaluated at discrete points in a program life cycle, while the leading indicators are designed for continuous evaluation throughout a program. The SERPR tools are intended to assess the qualities that should be inherent in effective Systems Engineering, and the Systems Engineering Leading Indicators measure how effectively those Systems Engineering tasks are being performed. Many of the qualities of effective Systems Engineering, as described in SERPR tools, are necessary to accomplish the work measured by each of the Leading Indicators.

Therefore, the mapping between Systems Engineering Leading Indicators and SERPR tools are not exact. The mapping shows the critical success factors that would be most influential in affecting the corresponding Leading Indicator measurements.

By researching this mapping, two leading indicators (System Definition Change Backlog and Facility & Equipment Availability) were identified as areas which appear not to be fully evaluated by the SERPR tools. This is an opportunity for future SERPR development.

For more information, see http://csse.usc.edu/csse/TECHRPTS/2009/usc-csse-2009-518/usc-csse-2009-518.pdf

Table C1 - Critical Success Factors Related to SE-Program Risks

High-level Goals	Critical Success Factors	
Concurrent definition of system requirements & solutions	1.1	Understanding of stakeholder needs
	1.2	Concurrent exploration of solutions
	1.3	System scoping & requirements definition
	1.4	Prioritization & allocation of requirements
System life-cycle organization, planning & staffing	2.1	Establishment of stakeholder life-cycle responsibilities, authorities, and accountabilities
	2.2	Establishment of integrated product team's life-cycle responsibilities, authorities, and accountabilities
	2.3	Establishment of resources to meet objectives
	2.4	Establishment of selection, contracting, and incentive structures
	2.5	Assurance of necessary personnel competencies
Technology maturing & architecting	3.1	COTS/NDI evaluation, selection, validation for maturity & compatibility.
	3.2	Life-cycle architecture definition & validation
	3.3	Use of prototypes, models, etc. to validate maturity
	3.4	Validated budgets & schedules
Evidence-based progress monitoring & commitment reviews	4.1	Monitoring of system definition
	4.2	Monitoring of feasibility evidence development
	4.3	Monitoring/assessment/re-planning for changes
	4.4	Identification and mitigation for feasibility risks
	4.5	Reviews to ensure stakeholder commitment

Table C2 - Mapping of Systems Engineering Leading Indicators to Critical Success Factors Related to SE-Program Risks

Leading Indicator	*Related SERPR CSFs*	
Requirements Trends	1.1 1.3 1.4 4.1	Understanding of stakeholder needs System scoping & requirements definition Prioritization & allocation of requirements Monitoring of system definition
System Definition Change Backlog Trend	4.3	Monitoring/assessment/re-planning for changes
Interface Trends	1.3 3.2 4.1	System scoping & requirements definition Life-cycle architecture definition & validation Monitoring of system definition
Requirements Validation Trends	1.1 1.2 1.3 3.3	Understanding of stakeholder needs Concurrent exploration of solutions System scoping & requirements definition Use of prototypes, models, etc. to validate maturity

Table C2 - Mapping of Systems Engineering Leading Indicators to Critical Success Factors Related to SE-Program Risks

Leading Indicator	***Related SERPR CSFs***	
Requirements Verification Trends	1.1 1.2 1.3 1.4 4.1	Understanding of stakeholder needs Concurrent exploration of solutions System scoping & requirements definition Prioritization & allocation of requirements Monitoring of system definition
Work Product Approval Trends	3.2 4.3 4.4	Life-cycle architecture definition & validation Monitoring/assessment/re-planning for changes Identification and mitigation for feasibility risks
Review Action Closure Trends	4.2 4.3 4.4	Monitoring of feasibility evidence development Monitoring/assessment/re-planning for changes Identification and mitigation for feasibility risks
Risk Exposure Trends	1.4 3.3 4.1 4.2 4.3 4.4	Prioritization & allocation of requirements Use of prototypes, models, etc. to validate maturity Monitoring of system definition Monitoring of feasibility evidence development Monitoring/assessment/re-planning for changes Identification and mitigation for feasibility risks
Risk Treatment Trends	3.2 4.1 4.2 4.3 4.4	Life-cycle architecture definition & validation Monitoring of system definition Monitoring of feasibility evidence development Monitoring/assessment/re-planning for changes Identification and mitigation for feasibility risks
Technology Maturity Trends	1.1 1.2 3.1 3.3 4.3	Understanding of stakeholder needs Concurrent exploration of solutions COTS/NDI evaluation, selection, validation for maturity & compatibility. Use of prototypes, models, etc. to validate maturity Monitoring/assessment/re-planning for changes
Technical Measurement Trends	3.1 3.2 3.3 3.4 4.2 4.3	COTS/NDI evaluation, selection, validation for maturity & compatibility. Life-cycle architecture definition & validation Use of prototypes, models, etc. to validate maturity Validated budgets & schedules Monitoring of feasibility evidence development Monitoring/assessment/re-planning for changes
Systems Engineering Staffing & Skills Trends	2.2 2.4 2.5 4.1 4.2 4.3	Establishment of integrated product team's life-cycle responsibilities, authorities, and accountabilities Establishment of selection, contracting, and incentive structures Assurance of necessary personnel competencies Monitoring of system definition Monitoring of feasibility evidence development Monitoring/assessment/re-planning for changes
Process Compliance Trends	2.1 2.4 3.2 4.1 4.3	Establishment of stakeholder life-cycle responsibilities, authorities, and accountabilities Establishment of selection, contracting, and incentive structures Life-cycle architecture definition & validation Monitoring of system definition Monitoring/assessment/re-planning for changes
Facility and Equipment Availability Trends	2.2	Establishment of integrated product team's life-cycle responsibilities, authorities, and accountabilities

Table C2 - Mapping of Systems Engineering Leading Indicators to Critical Success Factors Related to SE-Program Risks

Leading Indicator	***Related SERPR CSFs***	
Defect/Error Trends	1.2	Concurrent exploration of solutions
	4.1	Monitoring of system definition
	4.2	Monitoring of feasibility evidence development
	4.3	Monitoring/assessment/re-planning for changes
	4.4	Identification and mitigation for feasibility risks
System Affordability Trends	1.1	Understanding of stakeholder needs
	1.3	System scoping & requirements definition
	2.1	Establishment of stakeholder life-cycle responsibilities, authorities, and accountabilities
	2.3	Establishment of resources to meet objectives
	4.1	Monitoring of system definition
	4.2	Monitoring of feasibility evidence development
	4.3	Monitoring/assessment/re-planning for changes
	4.4	Identification and mitigation for feasibility risks
Architecture Trends	1.1	Understanding of stakeholder needs
	2.4	Establishment of selection, contracting, and incentive structures
	3.2	Life-cycle architecture definition & validation
	4.2	Monitoring of feasibility evidence development
	4.4	Identification and mitigation for feasibility risks
Schedule and Cost Pressure	2.1	Establishment of stakeholder life-cycle responsibilities, authorities, and accountabilities
	3.4	Validated budgets & schedules
	4.1	Monitoring of system definition
	4.2	Monitoring of feasibility evidence development
	4.3	Monitoring/assessment/re-planning for changes

APPENDIX D – Research Partners

D.1 Lean Advancement Initiative (LAI)

The Lean Advancement Initiative (LAI) at Massachusetts Institute of Technology (MIT) is an open consortium of key government, industry, and academic members. LAI researches, develops, and promulgates practices, tools, and knowledge that enable and accelerate enterprise transformation, and promotes cooperation at all levels and facets of an enterprise to eliminate traditional barriers to improving industry and government teamwork. LAI combines unique knowledge, product, and tool creation with practical deployment support to help its members address their priorities of: transformation, change management, product and service development, and systems engineering. LAI rapidly deploys new knowledge through knowledge exchange events, focused transformation deployment activities, LAI Educational Network curricula development, and training.

LAI was the initiating organization for the systems engineering leading indicator's project, following the 2004 Air Force/LAI Workshop on Engineering for Robustness. LAI is presently the co-lead for the project working with INCOSE and PSM, and in collaboration with other organizations. Many of the LAI consortium members have implemented the systems engineering leading indicators based on the guide. LAI has also sponsored research including one master thesis that examined the implementation of the indicators in an aerospace company. LAI runs periodic Knowledge Exchange Events (KEEs) to educate practitioners on the leading indicators, and has featured the effort in conferences and research summits in support of transitioning this work to practice.

For more information on LAI, please see http://lean.mit.edu.

D.2 Practical Systems and Software Measurement (PSM)

Practical Software and Systems Measurement (PSM) was developed to meet today's software and system technical and management challenges. It is an information-driven measurement process that addresses the unique technical and business goals of an organization. The guidance in PSM represents the best practices used by measurement professionals within the software and system acquisition and engineering communities.

PSM:
- Is sponsored by the Department of Defense and the U.S. Army
- Provides Project Managers with objective information needed to successfully meet cost, schedule, and technical objectives
- Is based on best measurement practices of DoD, government and industry programs
- Is a flexible process tailored to software and system processes
- Defines an information-driven analysis approach
- Supports current software and system acquisition and measurement policy
- Provides a basis for enterprise level management
- Is compatible with the ISO/IEC and IEEE 15939 standard, Software and Systems Engineering - Measurement Process

For more information, see http://eee.psmsc.com.

D.3 Systems Engineering Advancement Research Initiative (SEAri)

The Systems Engineering Advancement Research Initiative (SEAri) at the Massachusetts Institute of Technology (MIT) seeks to advance the theories, methods, and effective practice of systems engineering applied to complex socio-technical systems through collaborative research. SEAri conducts both theoretical and empirical field research to ensure rigorous and relevant contributions toward prescriptive research outcomes. The group fosters dialogue among senior system leaders across multiple domains and application sectors, and conducts research summits to share research progress and outcomes. SEAri leverages the resources of MIT and its strategic partners, and contributes to the body of knowledge through journal and conference publications. SEAri also develops curricula, education materials, and handbooks to inspire, inform, and guide students and practitioners. SEAri engages faculty, research staff, graduate and undergraduate students, and sponsors in collaborative research.

SEAri works in collaboration with LAI to perform research on leading indicators, and to transition the research to practice. During the past several years, SEAri has undertaken two areas of research related to systems engineering leading indicators. The first has examined the applicability of the leading indicators for systems of systems (SoS), including the identification of 'soft indicators'. The second is graduate research sponsored by the US Air Force Human Systems Integration Office to develop and enhance leading indicators, with related interpretation and use guidance, in support of human systems integration efforts. SEAri developed an appendix for this version of the guide. SEAri has featured these efforts in conferences and research summits in support of transitioning this work to practice.

For more information on SEAri, please see http://seari.mit.edu.

D.4 International Council on Systems Engineering (INCOSE)

The International Council on Systems Engineering (INCOSE) is a professional society founded to develop and disseminate the interdisciplinary principles and practices that enable the realization of successful systems. INCOSE's mission is to foster the definition, understanding, and practice of world class systems engineering in industry, academia, and government. INCOSE's vision is to be the world's premier professional society for advancing the art and practice of systems engineering.

This vision comes with a commitment to shaping a future where systems approaches are preferred and valued in solving problems, whether enabling holistic solutions to global challenges or providing solutions for product development issues. INCOSE makes this vision a reality through its members. Together, the organization supports a vision of the future of systems engineering focused on solving the tough problems encountered in technical and social system domains.

Learn more about what INCOSE can do for you at http://www.incose.org.

D.5 Systems Engineering Research Center - University Advanced Research Consortium (SERC_UARC)

The mission of the Systems Engineering Research Center is to enhance and enable the DoD's capability in Systems Engineering for the successful development, integration, testing and sustainability of complex defense systems, services and enterprises. To this end, SERC will operate as the systems engineering research engine for the DoD, responsible for identifying, evaluating, creating and integrating methods, and processes and tools that support effective systems engineering practice in the acquisition of weapons platforms, major defense systems, systems of systems, network-centric systems, and enterprise systems.

The SERC makes use of the SE Leading Indicators of this guide as a best practice for tracking and forecasting program success.

SERC efforts focused on the early identification of program risks are discussed in Appendix C - Early Identification of SE-Related Program Risks. In the appendix, SERC researchers compare the leading indicators with identified Systems Engineering related program risks. It is hoped that conclusions may be folded into the next revision of this guide.

For more information regarding the SERC, see http://www.sercuarc.org.

APPENDIX E - ACRONYMS

AoA	Analysis of Alternatives
AMA	Analysis of Material Approaches
CDR	Critical Design Review
DoD	United States Department of Defense
ICD	Initial Capabilities Document
INCOSE	International Council on Systems Engineering
KPP	Key Performance Parameter
LAI	Lean Advancement Initiative
LMCO	Lockheed Martin Corporation
MIT	Massachusetts Institute of Technology
MOE	Measure of Effectiveness
MOP	Measure of Performance
NAVAIR	Naval Air System Command
NDIA	National Defense Industrial Association
NGC	Northrop Grumman Corporation
PDR	Preliminary Design Review
PSM	Practical Software & Systems Measurement
RFC	Request for Change
SAIC	Science Applications International Corporation
SE	Systems Engineering
SEAri	Systems Engineering Advancement Research Initiative
SED	Systems Engineering Division
SEMP	Systems Engineering Management Plan
SEP	Systems Engineering Plan
SoS	System of Systems
SRR	System Requirements Review
SSCI	Systems and Software Consortium, Incorporated
TBD	To Be Determine
TBR	To Be Resolved
TPI	Technical Performance Index
TPM	Technical Performance Measure(ment)
V&V	Verification & Validation
WG	Working Group

APPENDIX F – DATA COLLECTION PROCEDURES

The following information is very organization or project dependent and will not be defined in this guidance. It is provided in this one indicator (Requirements Growth) as an example only. The organization or project measurement plans should include this information following the guidance of PSM.

Data Collection Procedure (for each Base Measure) ***Complete this section for each base measure listed in each measurement information specification***	
Frequency of Data Collection	Collect at least monthly; more frequently during peak activity periods. Do not sample - collect all requirements data.
Responsible Individual	Measurement Analyst, Requirements Manager, Configuration Management Manager
Activity in which Collected	From concept and system definition through system deployment
Potential Sources of Data	Requirements Database, Change Board records, defect data
Typical Tools Used in Data Collection	Requirement Database, Configuration Management Database
Verification and Validation	Check data against Configuration Management records.
Repository for Collected Data	User defined.
Data Analysis Procedure (for each Indicator)	
Frequency of Data Reporting	Biweekly to monthly, depending on the level of activity
Responsible Individual	Measurement Analyst
Activity in which Analyzed	From concept and system definition through system deployment
Source of Data for Analysis	Requirements Database, Change Board records, defect data
Tools Used in Analysis	Spreadsheet, statistical analysis, measurement analysis
Review, Report, or User	Chief SE, Product Manager.

Measurement Specifications

The table below describes the typical anatomy of the information measurement specification. The format of each leading indicators specification follows.

{Name of Leading Indicator}	
Information Need Description	
Information Need	*Specifies what the information need is that drives why we need this leading indicator to make decisions*
Information Category	*Specifies what categories (as defined in the PSM) are applicable for this leading indicator (for example, schedule and progress, resources and cost, product size and stability, product quality, process performance, technology effectiveness, and customer satisfaction)*
Measurable Concept and Leading Insight	
Measurable Concept	*Defines specifically what is measurable*
Leading Insight Provided	*Specifies what specific insights that the leading indicator may provide in context of the measurable concept - typically a list of several or more*
Base Measure Specification	
Base Measures	*A list of the base measures that are used to compute one or more leading indicators - a base measure is a single attribute defined by a specified measurement method*
Measurement Methods	*For each base measure, describes the method used to count the base measure, for example simple counting or counting then normalized*
Unit of Measurement	*Describes the unit of measure for each of the base measures*
Entities and Attributes	
Relevant Entities	*Describes one or more particular entities relevant for this indicator – the object is to be measured (for example, requirement or interface)*
Attributes	*Lists the subset of particular attributes (characteristics or properties) for each entity that are of interest for this leading indicator*
Derived Measure Specification	
Derived Measure	*Describes one or more measures that may be derived from base measures that will be used individually or in combination as leading indicators*
Measurement Function	*The function for computing the derived measure from the base measures*
Indicator Specification	
Indicator Description and Sample	*A detailed specific description and display of the leading indicator, including what base and/or derived measures are used*
Thresholds and Outliers	*Would describe thresholds and outliers for the indicator; this information would be company (and possibly project) specific*
Decision Criteria	*Provides basic guidance for triggers for investigation and when possible action to be taken*
Indicator Interpretation	*Provides some insight into how the indicator should be interpreted; each organization would be expected to tailor this*
Additional Information	

{Name of Leading Indicator}	
Related Processes	*Lists related processes and sub-processes*
Assumptions	*Lists assumptions for the leading indicator to be used, for example, that a requirements database is maintained*
Additional Analysis Guidance	*Any additional guidance on implementing or using the indicators*
Implementation Considerations	*Considerations on how to implement the indicator (assume this expands with use by organization)*
User of Information	*Lists the role(s) that use the leading indicator information*
Data Collection Procedure	*Details the procedure for data collection*
Data Analysis Procedure	*Details the procedure for analyzing the data prior to interpretation*