Systems Engineering Leading Indicators for Assessing Program and Technical Effectiveness

Donna H. Rhodes[1], Ricardo Valerdi[2], Garry J. Roedler[3]

[1,2] *Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA*

[3] *Lockheed Martin Corporation, EPI Center, Cherry Hill, NJ*

Email: rhodes@mt.edu

Abstract. This paper discusses a three year initiative to transform classical systems engineering (SE) measures into leading indicators, including the resulting guidance information that has been developed and future research directions. Systems engineering leading indicators are measures for evaluating the effectiveness of the systems engineering activities on a program in a manner that provides information about impacts that are likely to affect the system or program performance objectives. A leading indicator may be an individual measure, or collection of measures, that is predictive of future system performance before the performance is realized. Contrary to simple status oriented measures typically used on most projects, leading indicators are intended to provide insight into the probable future state, allowing projects to improve the management and performance of complex programs before problems arise. This paper discusses the motivations and collaborative development of the SE leading indicators. It defines the leading indicator construct, introduces the initial set of thirteen indicators, and provides guidance for implementation, analysis, and interpretation of these indicators. The initial set of indicators, developed through a collaboration of industry, government and academia, has recently undergone validation through pilot studies and surveys. This work serves as a foundation for industry implementation and for further research to improve and expand the set of indicators, including development of a better understanding of how to best implement and use the leading indicators in a given program context.

Keywords: systems engineering, measurement, leading indicators, metrics, effectiveness

1 Background and Motivation

Systems engineering measures and measurement practices formally made their way into industry practice in the late 1980's through mid-1990's (Rhodes et al 1988, Rhodes and Mastranadi 1992; Fisher 1993). These systems engineering measures largely extended from the measures and practices of the software engineering community (Grady 1987; PSM 1996) and the significant influence of capability maturity models (SEI 1997). During the 1990's, systems engineering organizations and professional societies made progress in defining useful measures and improving the measurement practices that included gathering, analysis, and reporting of measures[1]. This included

[1] In (INCOSE 1998), a measure is defined as "the result of counting or otherwise quantifying an attribute of a process, project or product. Measures are numerical values assigned to attributes according to defined criteria."

the addition of systems scope to the popular Practical Software Measurement (PSM 1996) to become Practical Software and Systems Measurement (PSM 2003), as well as development of the INCOSE SE Measurement Primer (INCOSE 1998).

This measurement guidance did not provide much guidance on how to analyze, interpret, and use the measures in a predictive manner, especially for measures targeted at systems engineering. Thus, most SE measurement has provided lagging information for systems engineering programs, but when gathered over time, was quite useful in showing history and progress of an organization's efforts. As the systems engineering community evolved in capability maturity (SEI 1995 and INCOSE 1996), the measurement programs and practices of organizations were more broadly implemented but their usefulness for real-time impact remained a challenge.

The concept of trend analysis, derivation of projections, and planning being improved by the availability of historical data is not a new concept (INCOSE 1998). Also, it has long been recognized that new projects can be budgeted, scheduled, and planned much more effectively if measures based on historical data from previous similar projects/products are used. However, the ability to use measurement information for predicative purposes within the context of a given program has remained an illusive practice. The INCOSE SE Measurement Primer (INCOSE 1998) defines the construct of a systems engineering leading indicator, however failed to provide any elaboration on how such a construct could be realized. It states that a leading indicator is: *(1) A measure or combination of measures that provides insight into an issue or concept. Indicators are often comparisons, such as planned versus actual measures, which are usually presented as graphs or tables. Indicators can describe the current situation (current indicators) or predict the future situation (leading indicators) with respect to an issue. (Adapted from PSM.) and (2) A mathematical composite of relevant, quantifiable, product, project progress or process attributes (measures) taken over time that communicate important information about quality, processes, technology, products, projects, and/or resources.*

The more recent impetus for the leading indicators project was the release of several policies calling for improved systems engineering on programs by the US Department of Defense and the US Air Force in 2003-2004[2]. During this period, the Lean Aerospace Initiative (LAI) research group at MIT was tasked with assisting with the systems engineering revitalization activity, and in June 2004, the *Air Force/LAI Workshop on Systems Engineering for Robustness* was held with over seventy leading systems engineers attending (Rhodes 2004). The workshop established the groundwork for several initiatives in support of systems engineering revitalization and the new challenges discussed in these policies. One of these initiatives called for *leading indicators for evaluating the goodness of systems engineering on a program*. This led to the formation of a collaborative working group with a charter for transforming traditional engineering measures into leading indicators that were more predictive in nature.

2 Collaborative Project to Develop SE Leading Indicators

In response to the US government policies on systems engineering revitalization and as triggered by the June 2004 Air Force/LAI Workshop, the SE leading indicators working group was initially formed under the

---

[2] 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

auspices of the Lean Aerospace Initiative (LAI). The working group, which was comprised of engineering measurement experts from industry, government and academia, also was formulated as a collaborative partnership with INCOSE[3] to draw from the wider systems community. The effort, co-led by two of the authors (Roedler and Rhodes), included measurement experts from LAI member organizations, INCOSE, and PSM[4]. The team held periodic meetings over the three year development effort, along with pilot programs, case study research, workshops, surveys, and tutorials. To ensure alignment with pre-existing measurement standards, the initiative used the information model from ISO/IEC/IEEE 15939. Measurement Process[5] and PSM to define the descriptive specifications for the systems engineering leading indicators. Original objectives for the leading indicators initiative were:

1. Gain common understanding of US Department of Defense needs and drivers of this initiative, and be tuned to industry needs;
2. Identify information needs underlying the application of SE effectiveness;
3. Identify a set of leading indicators for systems engineering effectiveness;
4. Define and document measurable constructs for highest priority indicators, including base and derived measures needed to support each indicator, attributes, and interpretation guidance;
5. Identify challenges for implementation of each indicator and recommendations for managing the implementation; and
6. Establish recommendations for piloting and validating the indicators before broad use.

Through a series of meetings, research, and drawing on the experience of many experts, a draft beta version of the SE Leading Indicators Guide was developed. The working group considered many different possible aspects of systems engineering that could be measured, and agreed on an initial set of thirteen leading indicators because these could replace basic systems engineering metrics presently in use. The working group decided to release the beta version first and work with various organizations to validate the indicators and complete the research and trials needed to build confidence in their intended usage. In December 2005, the beta release was issued and made available to the systems engineering community for trial use (Roedler and Rhodes 2005). It defined thirteen indicators:

1. Requirements Trends
2. System Definition Change Backlog Trend
3. Interface Trends
4. Requirements Validation Trends
5. Requirements Verification Trends
6. Work Product Approval Trends

---

[3] 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] 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.

[5] ISO/IEC/IEEE 15939:2007, Measurement Process is the current version of this standard. During the development of the SE Leading Indicators Guide, this standard was revised from its original version to broaden its scope from software to both systems and software.

7. Review Action Closure Trends
8. Risk Exposure Trends
9. Risk Handling Trends
10. Technology Maturity Trends
11. Technical Measurement Trends
12. Systems Engineering Staffing & Skills Trends
13. Process Compliance Trends

With the release of the beta version, the team began execution of a multi-faceted plan for the necessary validation of the indicators and refinement of the guide. Several organizations were pilot sites for the leading indicators, resulting in refinement particularly in regard to their implementation, analysis, and interpretation. An in-depth case study with investigation of the usefulness of the indicators in an organization was performed by an MIT graduate student (Flynn 2007). One of the participating corporations performed a corporate wide survey to gather expert opinions and experience of over 100 respondents regarding the relative usefulness of the leading indicators. All of these activities served to provide useful information toward refinement of the beta release of the guide in support of a Version 1.0 document. This period also involved a more extensive collaboration of between MIT, INCOSE, and PSM leveraging key events and other opportunities to gather feedback and recommendations.

In January 2007, LAI conducted a Knowledge Exchange Event at MIT in Cambridge, MA as a tutorial to share the work to date. The event was attended by thirty-seven experienced professionals from industry, government, and academia. Further feedback was gained, focusing on gathering inputs on interpretation, use, and possible pitfalls. This information was used in a subsequent meeting of the team, along with other validation results and studies, as inputs to the Version 1.0 guide.

The release of Version 1.0 of the guide (Roedler and Rhodes 2007) was announced at the 2007 INCOSE International Symposium, as a joint product of MIT, INCOSE, and PSM. The joint ownership is intended to promote wide use across the systems community and to obligate the respective organizations to work on its further evolution as a team, rather than diverge to create separate guidance documents. The philosophy for the guide is to promote and encourage organizational tailoring based on the common guidance, and over time to add appropriate appendices and information to improve its suitability for international use and its applicability for multiple domains and applications. Subsequent to release of the guide, several workshops gathered additional feedback and recommendations for the next version.

A second LAI sponsored tutorial was held in El Segundo, CA[6] in November 2007 with forty participants from government, industry and academia. The event resulted in the gathering of additional expert opinions, along with recommendations on how the indicators may drive positive and negative behaviors. Insights were also gathered on better ways to present the leading indicators, and on heuristics used by experts for technical and programmatic performance monitoring in context of each indicator. In addition, workshops were held as part of the PSM Users Group Conference in July 2007 and GEIA Engineering and Technical Management Conference in September 2007. These workshops resulted in several recommendations, including (1) need for training to

[6] The Aerospace Corporation provided hosting facilities to accommodate an LAI request for a US West coast event.

accompany the guidance; (2) need to look at additional leading indicators to address other common information needs, and (3) identification of various matrices for organizing the indicators by various relationships, for example, by program profiles or SE activities (Roedler 2007).

Figure 1 summarizes many of the activities leading to the Version 1.0 guide. The three year effort was purposively paced and designed to ensure adequate pilot use and collaborative development, both of which have been considered critically important to develop sound and tested guidance based on real experience and reviews by experts.

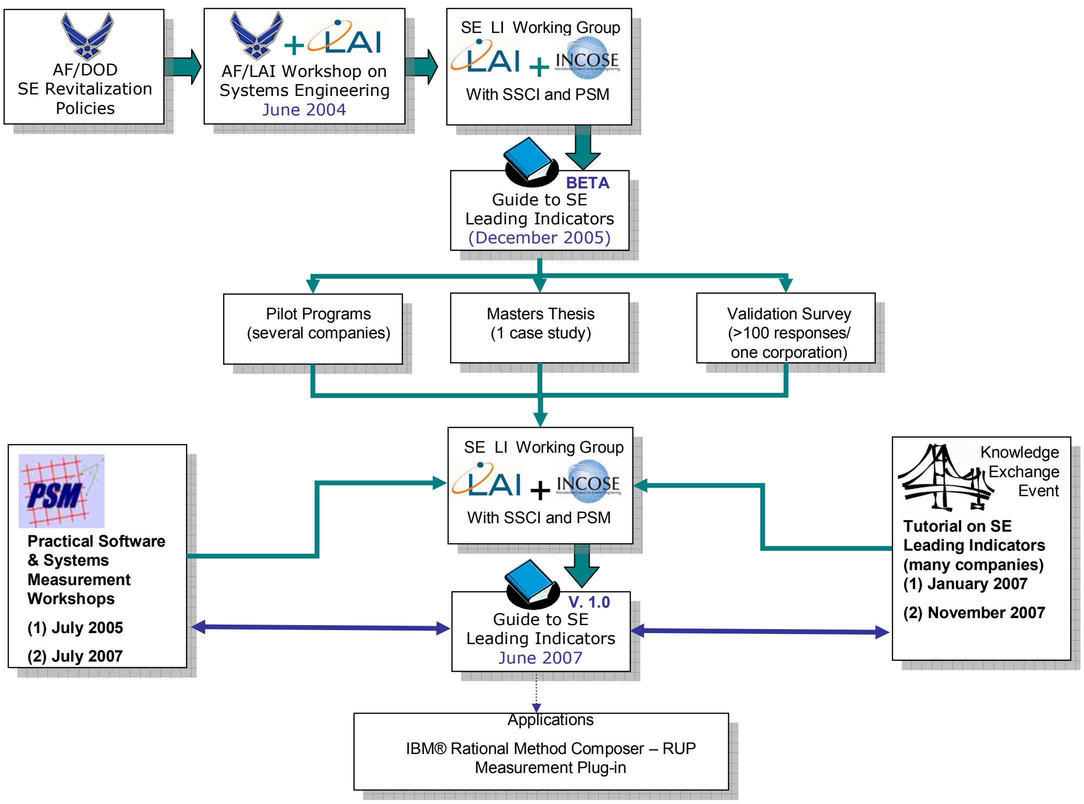


Figure 1. Three year initiative on leading indicators.

This project has been an excellent example of collaboration across industry, government and academic organizations. Although collaborative efforts such as this often take additional planning and time to accomplish, the positive outcomes and cross-organizational buy-in result in a superior result. This effort has been US-centric and aerospace-defense-centric due to the circumstances of its initiation, but the co-ownership and involvement of INCOSE provides the mechanism for seeking wider international involvement and diversity of perspectives as the guide evolves, along with the expertise to broaden the focus to commercial systems and product development. The wide review this guide received from the INCOSE review board has helped ensure broad applicability across systems engineering. While the authors believe the basic indicators can be adapted for wide use, they recognize that there is potentially a need for adaptation to fit the context of non-US and non-defense programs. The set of indicators has already seen a significant amount of adoption and recognition, including:

- Usage in the contributing corporations

- Adoption in some government programs
- Added to IBM® Rational Method Composer and RUP Measurement Plug-in (Ishigaki 2007)

3 Systems Engineering Leading Indicators

A leading indicator is a measure for evaluating the effectiveness of a how a specific activity is applied on a program in a manner that provides information about impacts likely to affect the system performance objectives (Roedler and Rhodes 2007). A leading indicator may be an individual measure, or collection of measures, predictive of future system performance before the performance is realized. Leading indicators are intended to aid leadership in delivering value to customers and end users, while assisting in taking interventions and actions to avoid rework and wasted effort. The goal of the indicators is to provide insight into potential future states to allow management to take action before problems are realized.

Conventional measures provide status and historical information, while leading indicators use an approach that draws on trend information to allow for predictive analysis. By analyzing the trends, predictions can be forecast on the outcomes of certain activities. Trends are analyzed for insight into both the entity being measured and potential impacts to other entities. Understanding the relationships between the indicator and other entities can help to predict probable states before they occur. An example of this type of analysis is illustrated in the PSM Analysis Model (PSM 2003) shown in Figure 2.

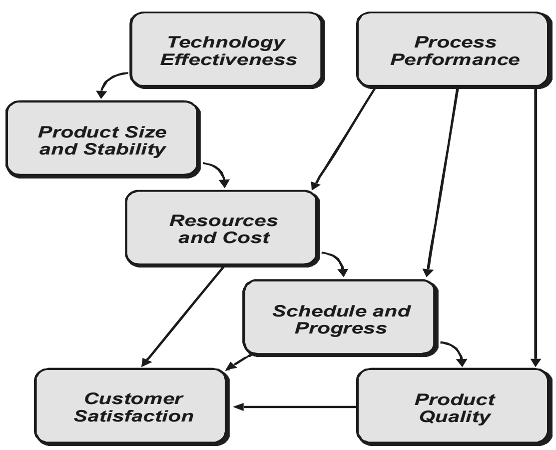


Figure 2. Issue Analysis Model Showing Relationships (PSM 2003)

Each of the arrows in the figure represents a relationship. For example, as Product Size and Stability changes (as observed by the leading indicator), so does Resources and Cost. Each of the common issue areas is related to its neighbor.

The indicators can provide engineering leaders with the information they need to make informed decisions and where necessary, take preventative or corrective action during the program in a proactive manner. While the leading indicators at first glance may 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*.

The motivation for measurement programs in general come from several sources, including internal motivations for programmatic and technical control, customer mandates for measurement to gain insight past performance and ongoing program performance, and motivations from competition and cooperation with peer organizations using measurement programs. The primary users are the program specific systems engineering leadership, program management, and IPT leadership who use the indicators to assess and make adjustments for assuring the systems engineering effectiveness of the program. Selected indicators may also be used by the program customer, program partners, and program suppliers depending on phase of program and nature of the contractual relationship. Secondary users include executive and business area management, as well as process owners, for the purpose of monitoring the overall effectiveness of systems engineering within and across programs.

The leading indicators are amenable to use in typical organizational measurement programs having an organizational measurement plan and a set of measures, augmenting the existing set of measures. For optimal efficiency the indicators should be tailored and implemented via the organization's measurement infrastructure (typically based on CMMI® practices or a similar industry standard), 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 information, expert opinion may be used as a proxy 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.

4 Anatomy of a Leading Indicator

Table I below describes the anatomy of the leading indicator measurement specification in regard to the various aspects of information that characterizes the indicator and its use. In Appendix 1, this information is detailed for one of the indicators, *requirements trends*. The guide provides a set of base information for each indicator, and it is expected that organizations using the guide will tailor and evolve this information, including adding organization specific knowledge in regard to interpretation and use.

Table I. Measurement Specification Template.

{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)*

{Name of Leading Indicator}	
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	*A detailed specific description and display of the leading indicator, including what base and/or derived measures are used*
Thresholds and Outliers	*Describes thresholds and outliers for the indicator; this information would be company (and possibly program) 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	
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*

4.1 Graphical Representation of a Leading Indicator

Figure 3 illustrates a very simplified, nominal graphic of the *Requirements Trends* leading indicator. The growth trend in the number of requirements with respect to planned number of requirements is typically based on expected value based on historical information of similar projects and the nature of the program. Based on actual data, a projected number of requirements can also be shown on the graph (it should be noted that this example is simplified and does not differentiate new, modified, and deleted requirements as actual program data will likely include). In this example, it can be seen that for the few months preceding Preliminary Design Review (PDR) that there is a significant variance in actual versus planned requirements, indicating a potential problem downstream. As requirements growth is an indicator of potential impacts to cost, schedule, and complexity of the technical solution, it was essential for the program team to determine when action is needed. An organization would then take corrective action – where we would expect to drive the actual growth back toward the planned values subsequent to this point. The requirements growth is an indicator of potential impacts to cost, schedule, and complexity of the technical solution. Requirements growth also indicates risks of change to and quality of architecture, design, implementation, verification, and validation. Understanding predictive relationships helps to interpret the indicator and make better decisions. Effective usage of the indicators would include reviewing results of these related measures concurrently with Requirements Growth. Some of the pitfalls of use and interpretation that may inhibit their effectiveness are discussed in section 5.1.1 and 5.1.2. It is expected that each organization will develop the graphical constructs for the selected indicators, with the appropriate levels of detail for the information in accordance with how the organization characterizes it. The leading indicators guidance document provides some basic illustrative examples to inform this activity, and it is expected that the systems engineering community will continue to evolve good examples of these over time.

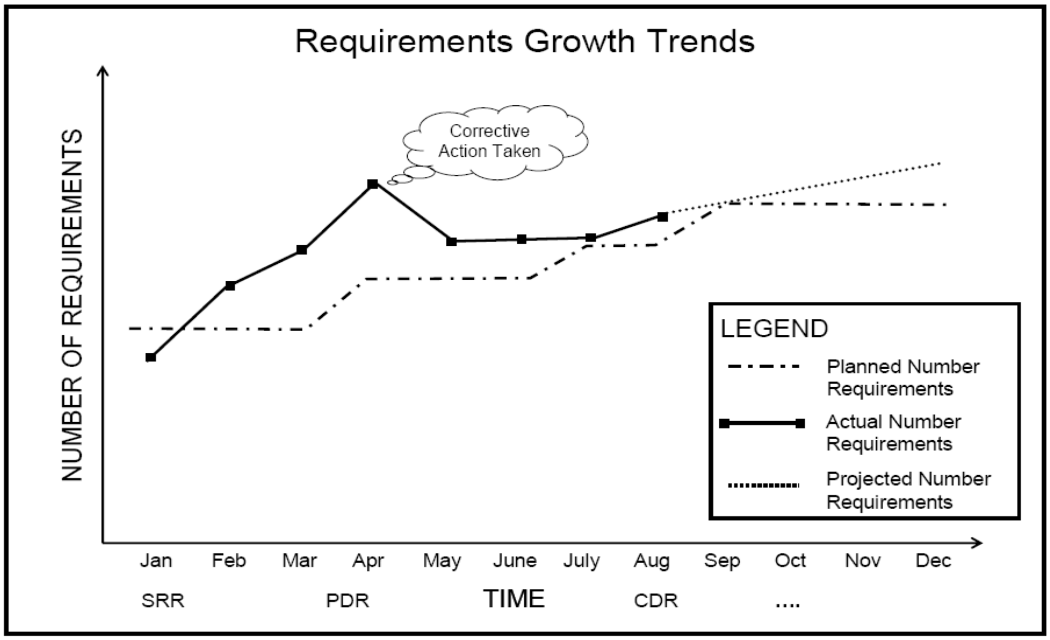


Figure 3. Requirements Growth Trends Leading Indicator

5 Interpretation and Use

The value of the leading indicators for a program lies with the insight that is provided. Meaningful insights permit program leadership to take appropriate action based on the impacts that may occur in regard to the systems engineering activity being measured by the indicator. Table II below describes the insight provided by the thirteen indicators described in the guide.

Table II. Insight Provided by Leading Indicator

Leading Indicator	Insight Provided
Requirements Trends	Rate of maturity of the system definition against the plan. Also characterizes stability and completeness of system requirements which could potentially impact design and production.
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 Handling 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

Leading Indicator	Insight Provided
	immature technology could introduce significant risk during development while failure to refresh dates technology could have operational effectiveness/customer satisfaction impact.
Technical Measurement Trends [7]	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	Ability of SE organization to execute total SE program as defined in the program SEP/SEMP. Includes quantity of SE personnel assigned, the skill and seniority mix and the time phasing of their application throughout the program lifecycle.
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 program risk.

Dialogue with systems practitioners provides a basis for identification of five top challenges for implementation of leading indicators on a program. All five of these implementation challenges motivate further work and research.

1. First, the validation of leading indicators as useful and appropriate for the activity being measured is essential; however it is difficult to get companies to share this. A few pilot programs have provided a preliminary validation, but this is insufficient and incomplete. As a result in the early phases of their use, the individual organization still should further validate the conditions under which a leading indicator can be most useful in their organization by observing its usefulness in various programs over time.
2. Second, due to similarity to classical metrics in the base measures collected, it may be difficult to get the management and practitioners in an organization to buy into using the leading indicators as they may on a superficial look be dismissed as 'something we are already doing'.
3. Third, it important to find effective ways to present the information in a concise and graphical form to address the specific information needs of the organization or project in order to aid effective decisions. At present, the guide has limited examples of this and future investigation is warranted.
4. A fourth challenge relates to the fact that interpretation of leading indicators is a blend of statistical analysis and subjective assessment based on experience. Some general interpretation guidance is provided in the guide, but there is a need to tailor to the project environment and experience. Thus, the interpretation guidance must be carefully developed with highly knowledgeable staff. This may very well represent a change in how a measurement program is conducted, and it may be difficult to get leadership support for using high level engineering talent for a 'task' that was previously performed by junior analysts. The effective use of the leading indicators requires a systems engineering program to not only gather and analyze the indicators, but to also dedicate experienced personnel's time to the interpretation of the resulting information. The ability to use the indicators for real-time management of programs requires decision making using this information at all levels.
5. The fifth challenge relates to using the leading indicators as a set. In some cases, it may be possible and necessary to find appropriate ways to aggregate leading indictors. It is equally important to know when aggregation should not be done, and also to know the relationships between the leading indicators. The selection of the subset of appropriate leading indicators in context of organizational factors is a rich area for future research.

[7] In addition to the guidance in the SE Leading Indicators Guide, a recent document entitled Technical Measurement Guide (Roedler, Jones 2005) provides greater depth on the selection, definition, implementation and analysis of TPMs.

The interpretation of measurement data requires gathering various forms of context information on the program and environment. A simple illustrative example is the *pizza metric*. Suppose that a program leader is tracking the number of empty pizza boxes found in an engineering laboratory on Monday morning, and based on experience knows that it is usually three to five boxes. On one given Monday, this leader finds that there are a dozen empty boxes. How should this be interpreted? Is the team so far behind that they worked all weekend and ordered extra pizza? Or, perhaps the team achieved a milestone ahead of schedule and had a celebration with invited colleagues? Or, perhaps Friday was 'bring your child to work day', and there were many extra people to feed yet the work and staffing was typical. The important point here is that the use of a leading indicator or metric as a standalone data point is a dangerous practice, and that only someone who understands the program nuances and its environment can interpret the leading indicator, and thereby discern if action needs to be taken when unanticipated values are seen in the data. Leading indicators provide important cues for making decisions, but not when used as 'standalone information'. As noted in (Flynn 2007) "it is not the intent of the indicators to deterministically declare the systems engineering efficiency bad or good, instead these are inputs into a decision process which requires the right people involved from the project to determine cause and course of action".

5.1 Pitfalls in Implementing and Interpreting Leading Indicators

Much of the literature on measurement warns us of the possible negative impacts of choosing certain measures because of the behavior these may create in organizations. The old adage of "what gets measured gets done" continues to hold true. In fact, some claim that the firm becomes what it measures (Hauser & Katz 1998). The leading indicators are no exception to this phenomenon. As such, an important part of the validation process for these indicators was to determine the possible pitfalls associated with their implementation and interpretation. In addition, the organization must ensure that the measures are not misused or "gamed" in order to satisfy reporting needs, competitive factors, or to avoid attention to the program. Each organization will need to develop appropriate methods and mechanisms to avoid such misuse.

The following summarizes the pitfalls for the Requirements Trends indicator that were identified during the validation process. For each indicator, the credible pitfalls or guidance to avoid the pitfalls has been examined and initial findings were incorporated into the Implementation Considerations section and the Interpretation Guidance section of the indicator specifications. Feedback from practitioners in industry and government helped to identify pitfalls in implementing and using this indicator.

5.1.1 Pitfalls in Implementation.

Considering the case of the Requirements Growth Trends Indicator (Figure 2) as an example, one pitfall has to do with the appropriate time intervals for measuring requirements. The desired frequency of measurement may not be available because the quantity of requirements is largely driven by milestone events such as design reviews and acceptance testing. Some organizations may be able to measure the requirements on a monthly basis but others may need to wait more than one year. Another pitfall associated with this indicator is the effort involved in counting requirements at a level where it is meaningful to the systems engineering organization. There needs to be some decomposition involved in the implementation of this indicator because the requirements as provided by the customer may not be as informative as the derived requirements. Finally, a standard definition for counting requirements is a necessary step and could lead to meaningless results if not defined accordingly. Deciding which requirements to count – functional, operational, environmental, etc. – is a critical step in ensuring that the right metrics are being tracked to indicate the progress of the systems engineering group.

5.1.2 Pitfalls in Interpretation.

Possible pitfalls in interpreting the *Requirements Trends* indicator include the context in which the trends are being evaluated. The volatility of requirements should be assessed in the context of the system life cycle and in relation to similar programs. Historical data and information can be helpful but it should be from programs that have not only product-based similarities but also customer-based similarities that reflect the way decisions were made about requirements. For instance, a program with multiple customers should not be compared to a program with only one customer simply because their trends of requirements over time may be significantly different. Some of the problems in the interpretation of trend analysis have been explored by Boehm et al (2003) in regard to a parametric cost models. The challenges in modeling cost and schedule are similar to the challenges in measuring leading indicators especially as they relate to the normalization of requirements and measurement reliability.

Another interpretation note is that the *Requirements Trends* indicator is closely associated with other leading indicators such as the *Requirements Validation & Verification Trends*. They may be more informative when used together. Additionally, the quantity of requirements is affected by additional system re-baselining. Users should keep this in mind when defining relevant thresholds associated with requirements volatility throughout the program life cycle. Finally, too much emphasis on requirements may lead to "level-mania" where there is too much management focus on raw data rather than on impact on continued progress. Requirements may not be adequate leading indicators on all programs. Users should determine which indicators are the best for their situation and incorporate similar measurement principles to manage progress.

Despite the amount of detail presented in these leading indicators, users should not assume that good performance in any particular indicator will lead to project success. Experience has shown that good performance on these leading indicators may lead to project success but there is little quantitative evidence that proves this correlation. In a survey of software 122 projects, Verner and Evanco (2005) found strong correlations between project success and goodness of requirements. While their data set was limited and their data collection method was survey-based, their study confirms the importance of requirements in project success. The Leading Indicators aim to measure whether an organization is doing the right things in terms of systems engineering. Even so, if systems engineering is performing well there are other aspects of the system that may cause the program to fail. The ultimate determinants of project success depend on the type of project and the particular organizations involved.

There are pitfalls in implementation and interpretation which include common ones for all indicators, as well as specific ones for each indicator. Program leaders need to be cognizant of these, and put in place appropriate strategies and mitigations.

6 Validation of the Indicators

The initial validation of the leading indicators has been accomplished through three mechanisms. These include (1) expert assessment of the usefulness of the indicators which has been gathered through a formal survey of one organization; (2) information gathering on pilot projects; and (3) experiences described in a case study of one organization. The positive impact of using the leading indicators was observed in several cases during the pilot program implementations. Expert opinions were also elicited and gathered in several workshops, and in an industry case study. The result of this early validation was a conclusion that the thirteen leading indicators defined in the beta version did provide for some new levels of insight into systems engineering effectiveness. As a result, all of the indicators from the beta level guide were elaborated and included in the version 1.0 guidance.

6.1 Expert Assessment of Usefulness of Indicators.

During the trial use period, there were several opportunities to gather expert assessment of the indicators. The implementation of the indicators on several programs in pilot studies provided information of their usefulness in context of the given program. Another assessment was gathered via a corporate survey in one organization. The survey was administered with responses from more than 100 chief engineers, technical leads, and program managers in key programs. The basis of that survey was to try to understand which of the indicators would provide utility and the right type of insight into programs. The respondents were asked to rate the indicators on a scale of 1 (least useful) to 5 (most useful) in terms of utility and all 13 scored fairly high; the lowest average score for any of the indicators was 3.9 (Roedler and Schimmoller 2007). Figure 4 shows the results of the survey of the usefulness of each indicator.

Indicator	Critical	Very Useful	Somewhat Useful	Limited Usefulness	Not Useful	Usefulness Rating *
Requirements Trends	24%	35%	11%	3%	3%	4.1
System Definition Change Backlog Trend	7	11	7	3	1	3.9
Interface Trends	14	12	4	0	1	4.3
Requirements Validation Trends	22	16	4	0	1	4.4
Requirements Verification Trends	37	23	6	2	1	4.4
Work Product Approval Trends	7	19	21	2	0	3.9
Review Action Closure Trends	5	33	21	5	0	3.9
Risk Exposure Trends	14	37	6	1	0	4.3
Risk Handling Trends	6	25	11	1	0	4.1
Technology Maturity Trends	6	6	7	0	0	4.1
Technical Measurement Trends	21	27	6	0	0	4.4
Systems Engineering Staffing & Skills Trends	11	27	15	0	0	4.2
Process Compliance Trends	6	14	11	1	0	4.0

Figure 4. Results of Organizational Survey on Usefulness of Indicators

6.2 Empirical Evidence of Positive Impact on a Program.

A good example of the positive impact of using leading indicators was demonstrated within one of the pilot projects, as shown in Figure 5 below. By monitoring the requirements validation trend, the program team was able to more effectively predict readiness for the System Requirements Review (SRR) program review milestone. Initially the program had selected a calendar date to conduct the SRR, but when the leading indicator showed that the requirements would not be validated by the date, leadership made the decision to have the SRR be event driven rather than by the preset calendar date. That is, the review date was set based on an acceptable level of requirements validation in accordance with the leading indicator. 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.

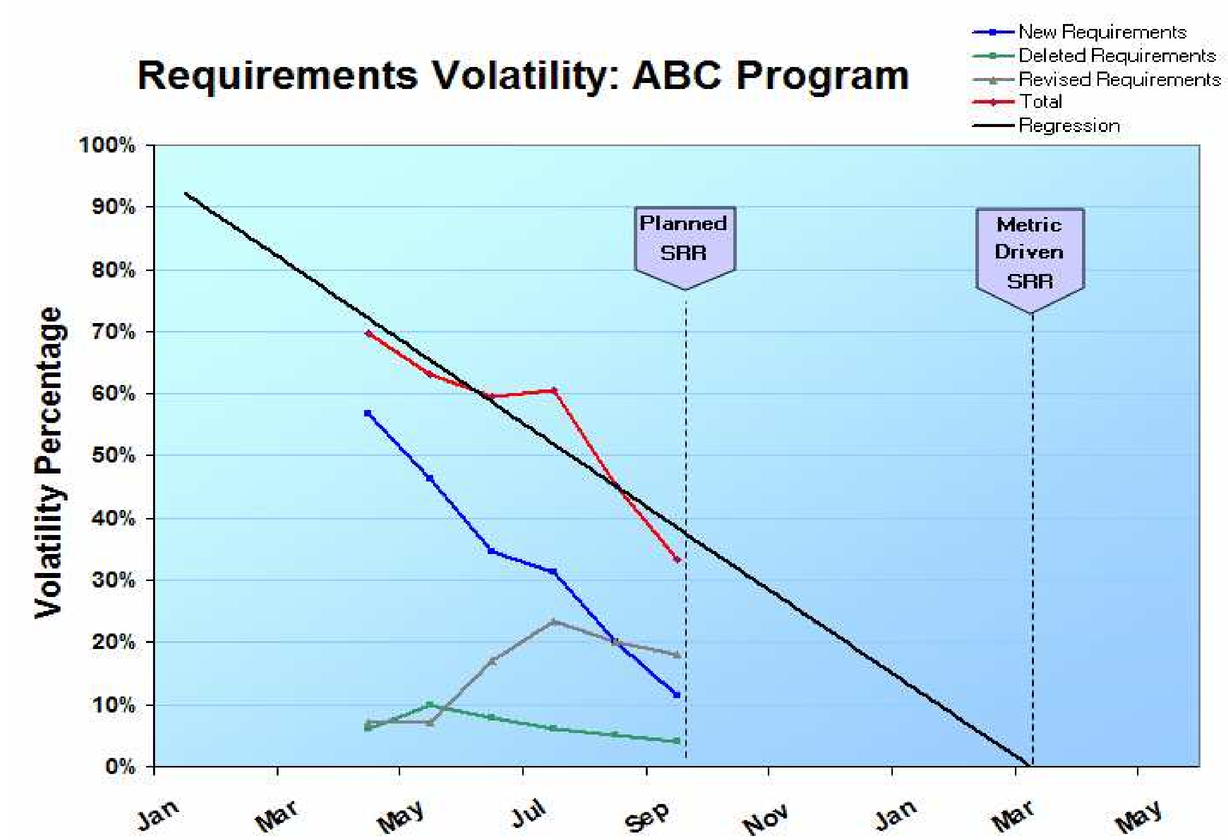


Figure 5. Using Leading Indicators for Program Decisions.

6.3 Insights Gained in Case Study.

During the development of the leading indicators, an in-depth case study was performed by a graduate student at an aerospace company (Flynn 2007). The goal was to evaluate the effectiveness with which a collection of 'prognostics' would forecast the efficiency of the application of systems engineering in a project and within an organization. Evaluation was performed through application of measurements to a case study of historical project data. This study was very insightful in understanding what issues teams encountered in implementing and using the leading indicators.

In an instance of examining the leading indicators for requirements trends, the study describes these as providing insight into the type of requirement changes occurring over the development phase. Flynn (2007) explains "Several new requirements were added late in this project. If an upward trend in requirements had been more visible to the entire project team and higher management it should have triggered an evaluation using the voice of the customer process employed during the development phase. This would have highlighted a missing project risk; the team assumed acquirer and user needs were relatively stabile as represented in the system performance specification provided with the Request for Proposal. It became evident late in the project this was not the case".

In addition to aspects purposively studied, this research study further cites "one of the issues unintentionally uncovered in this work was the inadequacy of existing historical data, even in organization rated CMMI level 3 or greater, to support setting of thresholds or reconstructing time histories of the salient features of the development phase without extensive interaction with the original project team. Far too many decisions made by the project team are retained as tribal knowledge, not documented in project specific artifacts, stored for knowledge sharing purposes or shared outside of the project tribe".

7 Future Research and Development

Several areas for future research have been defined, with a number of these already underway. These include: extension to other disciplines (e.g., human systems integration); extension of the set of leading indicators; applicability to system of systems engineering; elaboration and codification of heuristic information; and advanced leading indicators. Additionally, research is still needed to further pilot the indicators on real programs to study the impacts of decisions made in response to the request for research. A common piece of feedback is that training is needed for the indicators.

7.1 Extending the Leading Indicators.

Additional research is needed to extend the current set of leading indicators, and there are a number of possibilities that have been raised in feedback and review sessions with experts and in workshops. In the pilot phase case study in one aerospace company, the importance of early phase leading indictors was emphasized (Flynn 2007). "The concept generation phase lacks sufficient measurements to address the questions critical to determining the degree of concepts considered, the effectiveness of the selection process and the completeness of stakeholder needs elicitation. Some proposed measures include the number of concepts considered versus historical norms for organization, IPT effectiveness, number of concepts considered versus historical norms for organization, use of early phase simulation or set based design, experience of concept generation team, concept generation team dynamics, number of non-traditional concepts and measures of flexibility in the early architecture. The need for indicators that address the concept phase was also raised in two recent workshops.

In a recent workshop, participants indicated significant interest in a number of new leading indicators including (1) Concept Development indicators to provide early lifecycle feedback; (2) System of Systems Capabilities Trends (similar to tracking requirements trends); (3) Architecture Trends; (4) Algorithm Trends and Scenario Trends; (5) Complexity Change Trends (e.g., system, organization, etc. where changes in complexity that could impact cost, schedule, quality; (6) Resource Volatility, as amount of change in the resources required to support SE. Participants also discussed other possible leading indicators that were thought to be less useful. One of these was a Overarching SE Effectiveness Index which summarizes the other leading indicators. Although potentially useful, the experts were concerned about potential masking and the temptation to make decisions from a single number.   SE Productivity was cited as another less useful indicator. While of historical use, it was noted that productivity measures often are biased or misused.

7.2 Applicability to SoSE.

The leading indicators have primarily been targeted for traditional systems program, however there is potential for use on SoS programs. Some of the leading indicators are directly usable by a Lead Systems Integrator for an SoS program as indicators for SoS level engineering activities.   Additional thinking is needed around how leading indicators applied at the constituent systems level, by multiple systems in an SoS, could be used effectively as a collected set of indicators and/or as an aggregated indictor. It is recognized that lack of historical/baseline trend data for SoS level will be a challenge for implementing leading indicators, and further research is needed to validate the indicators on these types of programs. Additionally, some new leading indicators need to be evolved to address SoS specific activities.

7.3 Leading Indicators and Program Context.

An area of research of interest to practitioners would develop better knowledge on the indicators as related to the specific program contexts.   A useful area of inquiry would be to determine cost-effective sets of base measures that support greatest number of indicators, thereby allowing programs to minimize the effort in collecting data and maximizing the utility of the insight gained for effort applied. It has been recognized that there is not likely to be any "one-size-fits-all" solution but rather differences by type of program. Thus, these sets of measures could be best defined relative to program profiles.   Further, it was noted that a better characterization of the indicators versus program profile is desirable. Attributes should include size, customer type, contract type, application type (e.g., R&D, development, O&M, service mgt) and other factors. The indicators in the current version of the guide have been mapped to various stages and phases of the lifecycle. It has been noted that mapping the indicators to the SE activities would be very valuable particularly if this was at the process level and used a standard such as ISO/IEC 15288. However, there was a caution that the indicator set might become too large if this was done at too low a level of activity.

7.4 Using Indicators at Multiple Levels of Decision Making.

The systems programs today can be found at many levels, including product level programs, system level programs, and system of systems level programs. It is recognized that decision making at these levels is unique to some extent. As a result, there may be differences in the leading indicators themselves at the various levels, and certainly there will be differences in the interpretation and use at these different levels.   Case study research is needed to understand these differences so that leaders of different level systems can select appropriate indicators and use these effectively for decision making.

7.5 Heuristics as Step toward Prescriptive Guidance.

The participants in a recent tutorial session based on the leading indicators effort identified a number of preliminary heuristics as related to the thirteen indicators; an example heuristic is "Lack of completeness of interface definition and development can be an indicator of overall program risk". Additional case studies and pilot programs are needed to understand the various strategies and insights that leaders discover in regard to systems engineering effectiveness. Identifying such heuristics may eventually lead to prescriptive guidance.

7.6 Systems Engineering Leading Indicator Training.

Collective feedback is that there is a need to develop adequate training for these indicators to obtain a wide level of infusion across industry. This includes two levels of training, one for the decision makers and one for the measurement practitioners. The course for decision makers should be a short course focused on describing the SE Leading Indicators, the utility of the indicators, and the resources needed to implement them. The practitioner course should focus on selecting the right set of indicators, how to analyze and interpret for leading insight rather than lagging insight, and detail discussion and exercise for the indicators.

8 Summary

The paper has summarized the result of the three year effort to develop systems engineering leading indicators as a contribution toward more effective systems engineering practice. The Systems Engineering Leading Indicators Guide (Roedler and Rhodes 2007) provides useful guidebook with a detailed specification for each of the leading indicators as a foundation for organizations to use. The highly collaborative nature of the project, along with early validation through pilot programs, user surveys, and case studies, has resulted in new guidance that enhances the practice of effective systems engineering on programs. It is recognized that further application of the leading indicators is necessary and that additional leading indicators need to be developed. The authors propose three challenges in evolving this work: (1) data collection takes time and the systems community lacks a centralized repository to build a core history; (2) leading indicators require trend data and when an indicator is first introduced there is historical info on which to base trends; and (3) "next generation" leading indicators (concept phase and architecting are two key areas of interest) are important but much more difficult to define. The ongoing application within industry and government will provide important lessons and insights for further development of leading indictors, and the ongoing research will serve to extend the work in additional directions. The leading indicators effort is exemplary of the powerful outcomes that can be achieved when industry, government and academia collaborate on a strategic project.

ACKNOWLEDGEMENT. The authors wish to acknowledge the sponsorship of the Lean Aerospace Initiative[8]; International Council on Systems Engineering (INCOSE); Practical Software and Systems Measurement (PSM); and MIT Systems Engineering Advancement Research Initiative (SEAri). Special thanks to Paul Robitaille of INCOSE/Lockheed Martin and Cheryl Jones of US Army/PSM. Core team members included Ron Carlson, Dave Henry, Chris Miller, Mark Mithers, Rick Neupert, John Rieff, Sarah Sheard, Howard Schimmoller, Mike Ucchino, Gan Wang, and Bob Welldon. The leading indicators guide also includes the names of many other contributors to the beta guide and version 1.0 guide. A special acknowledgement goes to Dr. Marvin Sambur, former US Air Force Assistant Secretary for Acquisition for his efforts in motivating this work as part of the systems engineering

[8] LAI has recently been renamed the Lean Advancement Initiative (http://lean.mit.edu)

revitalization efforts. Contributions of many colleagues too numerous to cite include participants in several LAI, INCOSE, MIT, and PSM workshops, tutorials, and meeting, as well as pilot program participants.

REFERENCES

B. Boehm, D. Reifer, R. Valerdi, COSYSMO-IP: A Systems Engineering Cost Model, 1st Conference on Systems Integration, March 2003, Hoboken, NJ

G. Fisher, "Startup guide for systems engineering metrics*," Proceedings of the 3rd National Council on Systems Engineering*, Arlington, VA, July 1993, p 817-824

Grady, Robert B., Caswell, D.L., Software Metrics: Establishing a Company-wide Program, Prentice Hall, 1987

T. Flynn, Evaluation and Synthesis of Methods for Measuring System Engineering Efficacy within a Project and Organization, MIT Masters Thesis, February 2007

INCOSE, INCOSE Metrics Guidebook, *International Council on Systems Engineering*, July 1

D. Ishigaki, Integrating systems measurement into your software and systems delivery process with the RUP for PSM Plug-in, The Rational Edge, IBM, November 2007

ISO/IEC 15939:2007, *Measurement Process*, International Organization on Standardization/ International Electrotechnical Commission, 2007

E. Masselle, B. Radloff and D. H. Rhodes, "Putting a Yardstick to Systems Engineering, AIAA/IEEE Digital Avionics Systems Conference, 8th, San Jose, CA, Oct 17-20, 1988

PSM, Practical Software Measurement: A Guide to Objective Program Insight [PSM], Joint Logistics Commanders, Joint Group on Systems Engineering, Version 2.1, March 1996.

PSM, Practical Software and Systems Measurement: A Foundation for Objective Project Management [PSM], , Department of Defense and US Army, Version 4.0, March 2003.

D.H. Rhodes and A. Mastranardi, "Getting Started with a Measurement Program", *Proceedings of Second National Council on Systems Engineering,* Seattle, WA, 1992. pp 97-102

D.H. Rhodes, "Frequently Asked Questions on Systems Engineering Process", *Proceedings of the Fourth International Council on Systems Engineering*, San Jose, CA, August 1994, pp 263-269

D.H. Rhodes, A Wilbur, W. Miller, and R. Edelmann, "Metrics Concepts and New Directions"*, Proceedings of Fourth National Council on Systems Engineering*, Volume II, August 1994

D.H. Rhodes, A. Wilbur W. and W. Miller, **"**Metrics Working Group Report", *Proceedings of the Fifth National Council on Systems Engineering,* Volume II, July 1995

D. H. Rhodes W. Miller, **"**Metrics Progress Report", Proceedings of the Sixth International Council on Systems Engineering, Volume II, July 1996

D.H. Rhodes, Ed, Report on the AF/LAI Workshop on Systems Engineering for Robustness, Massachusetts Institute of Technology, Cambridge, MA July 2004

D.H. Rhodes, Leading Indicators of Program & Technical Performance for System of Systems, 2nd Annual Conference on System of Systems Engineering, July 2006

G. Roedler, D.H. Rhodes, et al, INCOSE SE Measurement Primer, International Council on Systems Engineering, March 1998

G. "SE Leading Indicators Guide", Proceedings of GEIA Engineering and Technical Management Conference, September 2007

G. Roedler and C. Jones, *Technical Measurement Guide*, International Council on Systems Engineering and Practical Software and Systems Measurement, December 2005

G. Roedler and D.H. Rhodes, (Eds), *Systems Engineering Leading Indicators Guide, Beta Version*, Massachusetts Institute of Technology, December 2005

G. Roedler and D.H. Rhodes, (Eds), *Systems Engineering Leading Indicators Guide, Version 1.0*, Massachusetts Institute of Technology, INCOSE, and PSM, June 2007

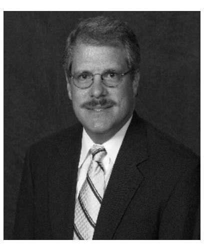
G. Roedler, H. Schimmoller, and G. Niemann, *Measurement Usage Document: Survey Results on Systems Engineering Leading Indicators*, Lockheed Martin Corp., EPI Document 280-17, January 2007

Software Engineering Institute*, Practical Software Measure: Measurement for Process Management and Improvement* [SEI MPM], Software Engineering Institute, Guidebook CMU/SEI-97-HB-003, April 1997

Software Engineering Institute, *A Systems Engineering Capability Maturity Model*^{SM}, Version 1.1, Software Engineering Institute, CMU/SEI-95-MM-003, November 1995
Verner, J. M., Evanco, W. M., In-House Software Development: What Project Management Practices Lead to Success?, IEEE Software, pp. 86-93, January/February 2005.

Donna H. Rhodes Dr. Donna H. Rhodes is a Senior Lecturer and Principal Research Scientist in the MIT Engineering Systems Division, and research director for the Systems Engineering Advancement Research Initiative (SEARI) at MIT. Her areas of specialization include technical and management theory and practices for architecting and design of complex systems, systems-of-systems, and enterprises. Prior to joining MIT, Dr. Rhodes had 20 years of experience in the aerospace, defense systems, systems integration, and commercial product industries. She received her PhD in Systems Science from SUNY Binghamton. Dr. Rhodes is a Past President and Fellow of INCOSE, and is a recipient of the INCOSE Founders Award.

Ricardo Valerdi Dr. Ricardo Valerdi is a Research Associate at the Lean Advancement Initiative (LAI) and the Systems Engineering Advancement Research Initiative (SEAri) at MIT and a Visiting Associate at the Center for Systems and Software Engineering (CSSE) at USC. He earned his BS in Electrical Engineering from the University of San Diego, MS and PhD in Industrial and Systems Engineering from USC. He is a Senior Member of the Technical Staff at the Aerospace Corporation in the Economic & Market Analysis Center. Previously, he worked as a Systems Engineer at Motorola and at General Instrument Corporation. He is on the Board of Directors of INCOSE.



Garry J. Roedler Mr. Garry J. Roedler is the Senior Program Manager of Systems Engineering (SE) at Lockheed Martin's Engineering Process Improvement Center. His responsibilities include managing corporate councils for SE, Test & Evaluation and Specialty Engineering to develop/select processes, implementation assets, training, and tools for the corporation. Previously, he led SE process improvement and achievement/sustainment of Level 5 CMM/CMMI objectives, including a world first to achieve Level 5 SE-CMM ratings. Garry has over 27 years experience in engineering, measurement, and teaching and holds degrees in mathematics education and mechanical engineering from Temple University. Other work includes leadership roles in various technical and standards organizations.