AD-A275 290

SYSTEMATIC MEASUREMENT

SPC-93071-CMC

VERSION 01.00.04

DECEMBER 1993

Robert D. Cruickshank
John E. Gaffney, Jr.
Richard Werling

Produced by the
SOFTWARE PRODUCTIVITY CONSORTIUM SERVICES CORPORATION
under contract to the
VIRGINIA CENTER OF EXCELLENCE
FOR SOFTWARE REUSE AND TECHNOLOGY TRANSFER

SPC Building
2214 Rock Hill Road
Herndon, Virginia 22070

SASET and PRICE are trademarks of Martin Marietta Corp.

CONTENTS

FIGURES

TABLES

ACKNOWLEDGMENTS

The authors wish to thank Bob Christopher, Ed Evers, Judd Neale, and Roger Williams for their insightful reviews of this document and the valuable commentary they provided. Dr. Andrew P. Sage of George Mason University did many penetrating reviews of this report, applying his knowledge of systems engineering to make the report far more incisive than it would have been without his contribution. Terry Snyder of Hughes Aerospace and Defense also provided a top-level review that gave this report a better focus. Mary Mallonee is owed a debt of thanks for her technical editing.

This page intentionally left blank.

EXECUTIVE SUMMARY

How can I avoid being blindsided by projects that are suddenly late, are over budget, or fail to satisfy customers' requirements?

How well do our development processes compare with those of our competitors with respect to time to market, quality, and productivity?

Answers to these questions can be provided by a systematic measurement program. Program benefits include better decision making, more precise control of program and project, cost avoidance by foreseeing problems and averting them, better management of risks, and measurable process improvement.

Management and measurement are inherently interconnected. The success of an organization is dependent on the quality of its management, and the quality of its management depends on the quality of the decisions made by management. Decision quality, in turn, is dependent on the quality of the information available. A measurement program that produces timely and meaningful information that supports management action is a program that enables decision quality.

This report provides the foundation for instituting a program of systematic measurement that supports enhanced organizational success. This success will occur because management will have the information needed to:

- Improve the predictability of product and system performance
- Increase competitiveness
- Improve the quality of delivered systems and products
- Improve the quality of the processes used to develop these systems and products
- Increase customer satisfaction
- Improve profitability

This report helps managers at all levels of the organization understand the need for and the mechanism of instituting a program for measurements to support software-intensive development projects. It shows how project information may be combined into a corporate-level knowledge base. In short, this report is a road map for adoption of a systematic measurement program at all levels of an organization concerned with development of software-intensive systems.

This report is addressed primarily to upper level managers with the profit and loss responsibility for one or more major software-intensive system development projects. It is this management level that has the authority to institute a program of systematic measurement, and it is this level that must be convinced of the management, technical, and economic benefits of such a program, although it is recognized that all levels of management have information needs and must "buy in" to the program.

A program of systematic measurement is a planned program that regularly measures the processes and products throughout the system development life cycle. The systematic measurement program is efficient in that it is designed to collect, process, and disseminate only the data needed to support management objectives. It includes the establishment of a database so that timely and action-oriented metrics are available to management for comparisons and decision making at all levels.

National and global forces are leading organizations to institute programs of systematic measurement. These forces range from the drive to comply with ISO 9000, the international standard on quality systems, to national initiatives for systems engineering standards from the Department of Defense, National Council on System Engineering, and Institute of Electrical and Electronic Engineers, and for process and capability maturity from the Software Engineering Institute. International competition in commercial development emphasizes "faster to market" technologies. These forces and initiatives require measurement to prove capability, competitiveness, and compliance.

Successful programs of systematic measurement exist at present, and these programs are producing the information that management requires for effective decision making. But, in many organizations, information is not available in the form of meaningful metrics for management to act upon. The data may be scattered over the wider organization and not be available in a timely fashion. However, there is great interest in measuring at the system level, i.e., at the level of the software-intensive system where hardware, software, procedures, and people and their interactions are considered. There is also interest in establishing systematic measurement programs to solve potential problems in the development of software-intensive systems before they become crisis situations.

In brief, this report:

1. Describes the competitive forces that push and pull management to measure their processes and products and shows how to develop meaningful information for timely decision making.

2. Shows that a program of systematic measurement supports the attainment of project and management goals. It enables planned collection, analysis, and application of project, process, and product data at the system development level. The enterprise that systematically measures process and products is the enterprise that maximizes both profits and customer satisfaction.

3. Demonstrates that the availability of meaningful and timely quantitative information and associated management metrics enables: (a) more precise estimation and budgeting; (b) better prediction of costs and schedule, thus preventing overruns; (c) better project tracking, including the anticipation of problems before they occur; and (d) increased capability to maximize customer satisfaction and quality through improved processes. These impacts of systematic measurement all lead directly to maximized profitability.

4. Quantifies the investment required by a systematic measurement program. One investment of $240,000 produced a return of 200%. Greater investment can produce returns on the order of 1,000%.

It is the hope of the authors and the Consortium that every organizational division, site, and profit center will conclude that the benefits of systematic measurement can greatly outweigh the costs and that they will institute such a program.

1. INTRODUCTION

Management information systems are still generally failing to meet their prime reason for existence. They fail to provide action-oriented information that enables responsible individuals to identify needs for action and that assists in accomplishing those actions.

Dick Werling, "Action-Oriented Information Systems"

1.1 BACKGROUND

The lack of timely, meaningful, and action-oriented information prevents management from making the kind of precise decision needed in managing its system development business. Lack of adequate informatior starts at the very bottom and percolates upward throughout the organization. What is required to solve this problem? The answer is a program of systematic measurement. This report defines systematic measurement at the system development level and illustrates how it can be applied to meet informati^n and decision-making needs at all levels with emphasis on the upper levels.

While managers may feel a lack of timely and meaningful information for decision making, the organization may be drowning in a sea of unstructured, unanalyzable data. Management in this situation may have no information that can be aggregated into a model for management, technical direction, and strategic decision making. To truly support management decision making, information must not only be available in a timely fashion, but it must be processed so that it has value for decision making.

The measurement of software products and software development processes has been a fact of organizational life for several years. Many software development organizations have instituted comprehensive metrics programs. Some of these programs are "systematic" in the sense that they are planned programs that measure software processes and products regularly throughout all phases of development at all organizational levels and over all software products. In hardware development, measurement has been mostly in the area of technical performance measures of the hardware product. More recently, because of the increased attention given to system-level planning, there have been proposals to measure at the system level, where a system is a combination of hardware, software, and people (including procedures). Each of these major system components of hardware, software, and people embodies a set of technologies. Because there is great interest in the effect of technologies on cost, development time, and product, measurement at the system development level is much to the point in today's competitive world. Measurement supports management at the system level and systems management of process and product as well.

This report presents and discusses the concept of measurement in system development and shows how measurement can be instituted at the system level through a program of systematic measurement. The primary purpose of the report is to induce upper level management to establish a program of systematic measurement, not just because it is a good idea, but because such a program will pay off in better management decisions, in cost avoidance, and in product quality enhancement.

1.2 WHAT IS SYSTEMATIC MEASUREMENT?

Systematic measurement develops meaningful information to support management decision making. It quantifies both system product characteristics and observable aspects of the system development process. More specifically, systematic measurement is the collection, analysis, and application of project, process, and product quantitative information at the system development level to support the attainment of project, system, and management goals. This information is valuable for supporting both project management and process improvement.

Systematic measurement is **an inherent part** of good management. It helps meet typical management goals for system development projects, such as:

- Stay within budget
- Meet all milestones (stay on schedule and complete ahead of schedule if possible)
- Meet and exceed process and product quality requirements
- Keep product development consistent with resource expenditure
- Meet and exceed product technical performance goals
- Minimize program and project risk

The measurement of processes and products at the system level cannot be partial or occasional; it must be applied to all processes and products from the system level down to the lowest level through the whole life cycle in a planned and efficient fashion. This type of measurement is called systematic measurement. When systematic measurement is applied to the whole system development project in a planned form through the whole life cycle, it is called a program of systematic measurement.

It should be noted that measurement by itself does not reduce costs, prevent overruns, or reduce risk. But the accumulated experience represented by measurement, especially if easily available in a database, enables better estimation and prediction. These metrics lead to reliable prediction, increased control of the development process, more precise management actions, reduced risk, and cost avoidance.

The prime goal of a systematic measurement program is better management, not just better measurement. A measurement program exists to serve the needs of management though the support it provides. It should also be noted that the systematic measurement program suggested in this report does not imply the existence of a "measurement czar." A properly instituted measurement program is established by upper level management but is accepted by all levels of management. An essential step in adopting a program of systematic measurement is obtaining the "buy in" of all levels of management. Management must make it clear to all employees that only process and product characteristics are measured; the measurement system is not a threat to employees.

This report uses the terms "measurement program" and "development project." A measurement program is the set of internal (to the organization, as directed by the senior manager) activities and responsibilities concerned with collection, analysis, and dissemination of quantitative information. A development project is the set of activities and responsibilities that is concerned with development of software-intensive systems. These terms are used consistently in these contexts in this report. It should be noted that a software-intensive system may be a subsystem of a larger system in the same way that the onboard flight control system is a subsystem of the larger air vehicle system (platform).

1.3 AUDIENCE FOR THE REPORT

The primary audience for this report is upper level managers who perceive that a lack of timely, meaningful, and action-oriented information is negatively affecting their business and decision making at all levels. Such managers usually have profit center responsibility for one or more major development projects involving software-intensive systems.

A secondary and nearly as important audience is the technical managers who also need information to make decisions about the development of software-intensive systems. This group will contribute information as part of the program of systematic measurement. A third group is the technical personnel who implement and execute the systematic measurement program.

1.4 BENEFITS OF THE REPORT

This report aids senior managers and the entire management chain in implementing a systematic measurement program that can help solve problems by providing timely and meaningful quantitative information about the system development process and the products it creates. The manager should be aware that a program of systematic measurement is affordable and will return benefits that exceed the investments in the measurement program. The management benefits of systematic measurement include better decision making, more precise program and project control, cost avoidance through problem anticipation and amelioration, better risk management, and measurable process improvement.

1.5 REPORT CONTENT AND ORGANIZATION

This report is composed of five sections:

- Section 1, Introduction, describes the objectives, benefits, and audience for the report and gives an overview of its contents. It defines the key role of measurement, both in improving control of software-intensive systems development projects and for improving the development process itself.
- Section 2, The Business Case for Systematic Measurement, describes some forces, external and internal to business organizations, that encourage the adoption of systematic measurement as a major tool for quantitative systems management. It also describes some case studies of system measurement.
- Section 3, Quantitative System Management, shows how to quantify requirements, activities, and processes required to develop software-intensive systems. A basic measurement and metrics set is described. The goal-question-metric (GQM) paradigm is described and applied to select specific metrics for implementation.
- Section 4, The Systematic Measurement Program, describes a systems development cycle and the basic metrics needed to support development of software-intensive systems and to support

the improvement of the process of creating such systems. It describes data requirements, methods for obtaining measures, the processing of raw data into meaningful action-oriented information, and the use of this information to support more effective management decisions including risk aversion.

- Section 5, The Adoption of a Program of Systematic Measurement, describes the general steps in the adoption of a systematic measurement program. It describes the adoption process, impediments to adoption, and how to cost effectively invest in a systematic measurement program.

- Appendix A, The State of the Practice of Systematic Measurement, presents two perspectives on the current state of the practice of systematic measurement: a survey on current system measurement programs and practices, as perceived by systems engineers attending the System Engineering Workshop organized by the Consortium (Software Productivity Consortium 1993a), and a perspective drawn from the Executive Round Table.

- Appendix B, The Information-Action Model, presents an information-action model for managers to judge the level of information maturity and the next stage of development in improving the information-action level of their own organization.

- Appendix C, Systems Engineering, describes technical management of the process used to develop software-intensive systems.

1.6 TYPOGRAPHIC CONVENTIONS

This report uses the following typographic conventions:

Serif font .... General presentation of information.

Italicized serif font .... Mathematical expressions and publication titles.

Boldfaced serif font .... Section headings and emphasis.

Boldfaced italicized serif font .... Run-in headings in bulleted lists.

2. THE BUSINESS CASE FOR SYSTEMATIC MEASUREMENT

What gets measured gets done.

Anonymous

2.1 EXTERNAL FORCES DRIVING ADOPTION OF SYSTEMATIC MEASUREMENT

Today, three types of external forces drive managers to consider adoption of systematic measurement:

1. Initiatives and standards promulgated by the federal government and national bodies
2. Pressures of world-wide competition and new international standards
3. Demands for profitability

These forces affect both developers of systems for commercial markets and suppliers to the Department of Defense (DoD), all of who must also wrestle with these forces.

2.1.1 Government and National Initiatives

2.1.1.1 Gov rnment Systems Engineering and Software Engineering Standards

Measurement is an implicit and powerful requirement of each of the standards discussed in this section. The DoD indicated its intention to reward provably better management in its contractor selection process. This management and implied measurement requirement applies to internal DoD software development and to nongovernment DoD-related contractor software developments. Good professional practice requires that it be applied to commercial software development projects. The measurements and metrics implicitly or explicitly required include:

- Cost, duration, and effectiveness of development process tasks
- Technical performance measures
- Technical and management parameters (metrics) for project tracking
- Technical parameters (metrics) for planning
- Metrics to quantify system and subsystem cost effectiveness

A primary initiative that affects the engineering of DoD systems is the new MIL-STD-499B (Draft), *Systems Engineering* (Department of Defense 1993b), the successor to MIL-STD-499A. This standard presents an updated framework intended to be applied to military development projects. It advocates "systems thinking" from the initiation of planning for the system through design, development, deployment and the eventual disposal of the system. This standard includes or implies such policies and practices as:

- Employment of the systems engineering process as the primary process for all development activities and tasks
- Integrated coincident design and development of products and systems, i.e., concurrent engineering (Sage 1992)
- Planning for systems engineering from the start through the Systems Engineering Management Plan
- Organizing the development project by an integrated product team (IPT), which is staffed on a multidisciplinary basis
- Use of event-based scheduling through the Systems Engineering Master Schedule
- Design and application of a project work breakdown structure (WBS) that is product oriented and relates the elements of work to be accomplished to each other and to the end product

Another standard that affects the engineering of software-intensive systems is DOD-STD-2167A, *Defense System Software Development* (Department of Defense 1988) and its planned successor MIL-STD-SDD (now 498) (Draft), *Software Development and Documentation* (Department of Defense 1992a). These standards establish uniform requirements for software development processes that are applicable throughout the system life cycle. Although they are intended as standards for development of defense software systems, they have been widely applied to development of nondefense systems.

Both *Department of Defense Directive 5000.1* (Department of Defense 1991a) and *Department of Defense Instruction 5000.2* (Department of Defense 1991b) demand measurement and validated data on systems development from contractors. These documents define procedures for acquiring and developing new defense systems. Their effects will guide future developments of new DoD programs.

2.1.1.2 Corporate Information Management

The Corporate Information Management (CIM) initiative, sponsored by the Office of the Secretary of Defense, is intended to improve the efficiency of business processes in DoD (Strassmann 1992). This begins at the level of business function, what the organization does. At the level of information resources, redesigned business processes are improved through more effective information handling. The technical support to this initiative is provided by many organizations within DoD.

The CIM initiative describes a functional improvement process. CIM process improvement, validation, and investment decisions demand measurement through the collection, analysis, and dissemination of timely data and meaningful metrics.

2.1.1.3 Defense Information Systems Agency

The Defense Information Systems Agency's CIM agency is intended to support DoD's CIM initiative throughout the DoD. This requirement affects both DoD and its contractors.

2.1.1.4 Software Engineering Institute

The Software Engineering Institute (SEI) has written extensively on the subject of software measurement. The material generated at the SEI, while technically not government standards, still carries the weight of standards in many cases. Measurement is a prime requirement of the SEI Capability Maturity Model (CMM), and much measurement capability is required for advancement from one CMM level to the next (SEI 1992; Paulk et al. 1993).

2.1.1.5 National Council on Systems Engineering

Several organizations, such as the National Council on Systems Engineering (NCOSE), are active in defining systems engineering. The work of the NCOSE metrics group covers current practices of systems engineering metrics, recommending a set of systems engineering metrics, best practices in the use of systems engineering metrics on real programs, advice for those starting up a metrics program in their own organizations, and how benchmark (best practices now in use) data relative to the systems engineering process and metrics can be used as a basis for suggested improvements. Measurement technology is obviously a prime component of all these areas of effort (NCOSE 1993).

2.1.1.6 Institute of Electrical and Electronic Engineers

The Institute of Electrical and Electronic Engineers (IEEE) has developed many standards, most of which require measurement (IEEE 1992). Standards most relevant to systematic measurement include: IEEE 1045–1983, Standard for Software Productivity Metrics; IEEE 982.1, Standard Dictionary of Measures to Produce Reliable Software; and IEEE 1028–1988, IEEE Standard for Software Reviews and Audits.

2.1.2 International Standards

New international standards, such as the International Standards Organization (ISO) set of ISO 9000 standards (International Organization for Standardization 1990) for quality management and quality assurance systems, require measures of both management and technical performance to demonstrate conformance. ISO 9001 applies to complete system products; ISO 9000–3 interprets ISO 9001 for software. Measurement is becoming a high priority requirement and activity, especially on high-reliability systems, such as embedded medical devices, X-ray equipment, telecommunications, and nuclear power systems. ISO 9000 certification is becoming a de facto requirement for doing business in the European community.

2.1.3 Competition

Systematic measurement is required to gain and maintain a competitive edge in international markets, which are strongly influenced by the following forces:

- Competitive cost pressures from less expensive overseas nations are now being felt in the United States. For example, U.S. organizations increasingly contract to Indian companies for software. Eastern European and Pacific rim nations are entering this arena as well. To meet such competition, it is necessary to control costs and to improve processes, and these objectives are dependent on measurement.

- Commercial markets have become more important with the decline of the defense market. Perceptions of commercial products have changed dramatically, with the consumer now demanding ever higher quality. Achieving higher quality demands an increased measurement of products and processes.

- Continuous process improvement practices, such as the Japanese Kaizen (Imai 1986) set of methods, are now used in many industries. Continuous process improvement practices are all based on measurement.

2.1.4 Higher Profitability

Measurement is associated with and leads to higher profitability by enabling:

- Better estimation and prediction of resource requirements
- Increased control of the development process
- Improved risk reduction or aversion
- More precise management actions
- Cost avoidance

Measurement is necessary but not sufficient for higher profitability. The prime requirement is not just better measurement but better management. It is not the case that more measurement must always result in higher profits because measurement, like any other activity, is subject to the law of diminishing returns. As management defines its own needs and goals, the correct level of measurement activity will become evident.

2.2 MEASUREMENT PROGRAM EXPERIENCES

This section presents some successful experiences with measurement programs.

2.2.1 Computer Manufacturer

In this actual experience, the site general manager of a large computer manufacturer needed more credible estimates and more reliable status reports on projects developing hardware and software for the DoD. A new "Software Cost Engineering" group solved his immediate problem by improving the reliability (accuracy) of the estimates for software cost proposals. Cost engineering also focused on more realistic budgeting of software development activities. The group determined that, in this site, budgeting for software development had always been significantly less than what was really needed. While no formal assessment was made, it was estimated that the software development organization was at SEI Process Maturity Level 3.

The cost engineering function was financed by overhead funds for a year. After the first year, the function received funds from each new development project and had to pay its way by helping project managers monitor their projects and by making cost estimates for new software development proposals. The function found that it could do its job well on an amount equal to an additional 2% to 3% of each software development contract. This arrangement was satisfactory to management.

The software cost engineering group soon realized that it was in the measurement business, not just in the cost business. Software product size, schedule, and quality estimates became issues of the same importance as software development cost. The group's accomplishments in the first 2 years were:

- Identifying each of the software development processes at the site and describing each of these processes in terms of the activities that composed it. A Software Engineering Process Group (SEPG) was formed some years later and used this information as its starting baseline.

- Quantifying each activity in each process by calculating the unit cost in labor months per 1,000 source lines of code (LM/KSLOC). This accomplishment involved investigating completed software development projects to learn actual sizes and costs for the process. With this information, management was better able to track projects and predict possible overruns.

- Convincing project management and the finance organization to create a WBS (Department of Defense 1992b, 1993a) for each software development project that accurately represented the development tasks to be done. This enabled better information to be collected for future estimates and also enabled more precise project monitoring.

- Inventing new and more accurate estimating procedures for software development cost, product size, and schedule.

- Creating a software measurement database with cost, size, schedule, documentation, and descriptive material about each new development project. This proved to be an invaluable resource for both estimators and managers. A software quality database was created at about the same time by the quality assurance organization. This database contained all the results of software code inspections on site and, when combined with the information from the measurement database, proved helpful in predicting defect densities of software during development. Better quality software resulted.

Working with the improved information provided by the software cost engineering group, the manager drove down the number of cost and schedule overruns in software development. The leverage was very high; the amount of cost avoidance was found to be much higher than the cost of the measurement program. The result was that systematic measurement was applied to all projects and all management activities. Table 1 shows the first 4 years' performance in predicting project cost.

2.2.2 Large Airframe Manufacturer

The measurement program in this actual experience is at the project level. The product is a large aerospace platform, involving simultaneous development of both hardware and software. The basic mode of tracking in this project is to monitor completions of a great number of milestones. Over the life of this project, 16,000 significant accomplishments are monitored. IPTs, each of which is involved with the development of major subsystems, get weekly and monthly reports that indicate compliance on the indicators relevant to them.

Table 1. Measurement Program Performance

Years Since Measurement Function Start	Software Development Project Completed in Year Shown	Percent Difference (Absolute) Actual Cost Versus Budget	Yearly Average Percent Difference
2	A	56.2	77.8
	B	99.4	
3	C	19.4	22.8
	D	31.4	
	E	14.3	
	F	33.6	
	G	15.2	
4	H	23.7	11.8
	I	2.2	
	J	9.6	

The principal classes of items monitored are technical performance measures, schedule attainment, and costs. The actual expenditure to date and the earned value measures are provided to IPT leaders through a cost and schedule control system. Cost and schedule estimates for hardware and software were developed originally by the pooled talents of experienced people with the right expertise.

The systematic measurement system employed by this project is considered successful because, to date, management has been made aware of cost and schedule problems in sufficient time to take steps to avert them before significant damage was done. Management believes that the magnitude of the benefit is a competitive advantage and will not provide details. However, it is clear that the cost savings from problem amelioration and avoidance is much greater than the cost of the measurement program. The primary value of the metrics collection, analysis, and reporting system briefly described here is that it ensures that all organizational units affected by a problem are made aware of that problem (divergence from satisfaction of technical requirement, schedule satisfaction, or cost objective) as quickly as possible. Thus, the system provides information to management to take action as expeditiously as possible.

2.2.3 Summary

The two measurement program experiences presented do not include a quantification of the benefits because the managers are reluctant to release financial and performance data of any kind (Cruickshank and Lesser 1982). However, the details of the programs are not as important as the management perception that these programs were successful and contributed to the successful conclusion of one or more development projects. The reader should conclude that a program of systematic measurement can contribute much value to the development of software-intensive systems. A successful program of systematic measurement contributes more value than it costs.

Appendix A presents the results of a survey of system measurement experiences and practices. Appendix A also contains a summary of the Executive Round Table on management metrics.

3. QUANTITATIVE SYSTEM MANAGEMENT

Many companies now recognize that their cost systems are inadequate for today's powerful competition. Systems designed mainly to value inventory for financial and tax statements are not giving managers the accurate and timely information they need to promote operating efficiencies and measure product costs.

R.S. Kaplan, "One Cost System Isn't Enough"

The quantitative system management (QSM) process is a framework for system management that integrates the concepts of management, system measurement, process improvement, and statistical process and quality control. QSM is managing efficiently "by the numbers." It employs quantitative information to drive products of the development process toward quantified goals and to incrementally assess the degree to which these goals are likely to be attained.

The basic concept of QSM was developed to be applied to software development (Software Productivity Consortium 1992), but the concept applies equally well to more broadly based system development.

3.1 OBJECTIVES OF QUANTITATIVE SYSTEM MEASUREMENT

Better visibility into the cost, schedule, effectiveness, technical performance measures, and quality of product development is needed for making effective project and process management decisions. There is generally no lack of data or lack of measurement. The lack is of relevant information needed for effective decisions. A program of systematic measurement provides this relevant information and associated project visibility. The following sections describe how system measurement data is collected, processed, and presented as information for both project control and process improvement.

- *Project Control.* Project control in the development of software-intensive systems involves periodically comparing the degree of actual achievement with preestablished plans and goals. It also involves taking the appropriate actions to mitigate effects of any problems that are discovered or anticipated.

- *Process Improvement.* An organization's system development process is improved by changing the methods, techniques, standards, and tools used in system development and support. This can result in improved products that exhibit higher quality at the same or lower cost than those created using the earlier process. Higher quality is demonstrated by lower defect levels and higher functional content relative to cost. "Cost" relates to the consumption of all relevant resources, including labor, money, and time.

3.1.1 The Quantitative System Management Process

QSM is a model of management action that includes:

- Setting process and product goals
- Measuring the system development project performance at selected points in time throughout the development process
- Analyzing the measurement data to discover trends and any existing or anticipated problems
- Determining risks
- Feeding back the findings in the form of corrective action recommendations
- Aggregating data from multiple project development experiences to aid in the improvement of the development process

Figure 1 shows schematically a process where the output of each process is the input to the next. This simple process is defined by approved methods and standards and has goals assigned by management, but it is a process with no measurements and no feedback to improve process performance.

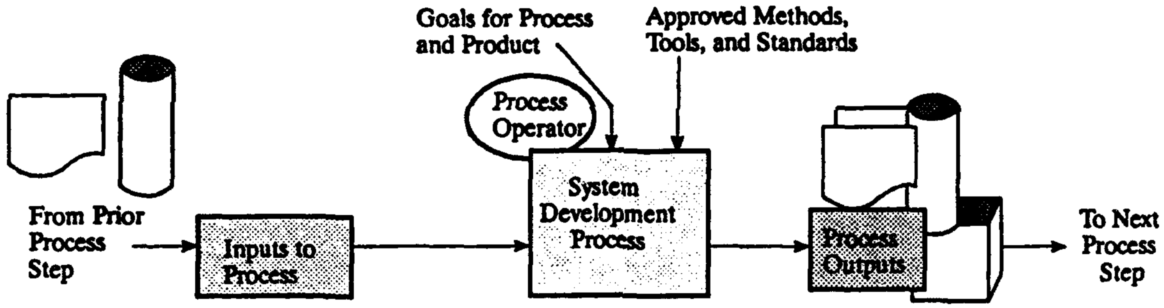

Figure 1. A System Development Process With No Feedback

Figure 2 shows how the process is improved by measurement and feedback of operational results. The process now has the means, i.e., feedback of information, for improvement of itself. Each piece of measurement data is a directly observable characteristic of a product or process that can be measured. In this system, early and more cost-effective visibility of problems means early corrective action. This characteristic is the most important aspect of QSM.

Figure 2 shows three users of information. The first user of information is the process operator (1). The second user, the process database (2) helps improve future estimates and process control. The third user is composed of various levels of management (3). In the figure, management actions are implemented as specific instructions, as improvements to the process, or as approved methods and standards.

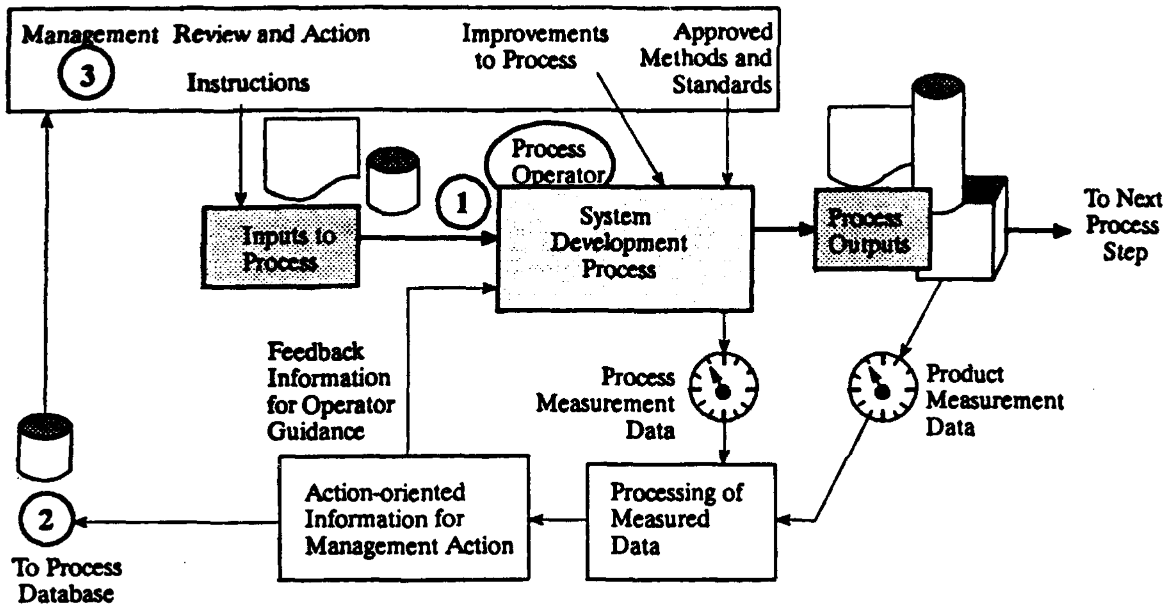

Figure 2. A System Development Process With Feedback

3.1.2 Results of Better Information

A systematic measurement program supports an organization's strategic objectives. In the face of competition, management can focus attention on improvements to:

- Shorten time to market
- Increase quality
- Decrease costs of development and deployment
- Translate customers' needs into requirements
- Manage cost, schedule, and technological risk
- Verify that products meet requirements

These results are based in part on measurements from past development projects and on the management metrics derived from this data. Practical development of the product, from mission objectives to implementation and test, is not purely mechanical. People need to apply their knowledge to interpret the measurements of the process. Involved people, such as the system engineers and the system development managers, use measurements and metrics to make estimates, predictions, and decisions. This means that system measurement data must be collected, analyzed, and available in appropriate form to aid in decision making.

Appendix B shows an "Information-Action" model that relates management actions and decision making to the availability and quality of information.

3.1.3 Basic System Metrics

A systematic measurement program involves the collection and analysis of system measurements on a regular planned basis. The main categories of system measurements are:

- Technical performance measures:
  - Functionality, reliability, maintainability, availability, etc.
  - Product size (weight, dimensions, number of components, etc.)
  - Quality, reliability
- Product size and cost (effort) for systems, subsystems, and documentation
- Process cost, effort, and schedule
- Risk

3.1.4 Selection of Metrics Using the Goal-Question-Metric Paradigm

The GQM paradigm (Weiss 1981; Basili and Weiss 1984) is a method that helps to specify metrics that support the development of software-intensive systems and that support the improvement of the process for creating such systems. GQM is a technique that selects only the metrics needed to support the project or management. GQM is an expression of efficient measurement; therefore, it is a control that prevents too much or too little measurement for management situations.

The power of the GQM paradigm is that it is a systematic method for selecting metrics appropriate to the needs of each information consumer. Also, GQM may help reduce the costs of data collection by concentrating data collection and analysis effort on only the metrics that are needed. Each metric has been clearly shown to support project goals, such as assessing the likelihood of attaining a product development objective of staying within a maximum cost bound. The metrics resulting from the application of the GQM paradigm quantify the characteristics of system products, processes, and development progress that are most useful to project management.

The GQM paradigm is a rule that states that system process and product measurements should be selected to support project and management goals. This means that measures should not be picked "out of the air." Rather, they are based on a perceived information need. The three principal steps of the GQM paradigm are:

1. State the **goal**. This answers the questions, "Who is the information consumer and what does he need to know?"

2. State the item of information that the consumer wants to know. This answers the **question**, "What question is the consumer going to ask to determine if the goal is satisfied?"

3. State the specific **metric** that you need and the things that are to be measured to answer each of the questions posed in step 2. This step answers the question, "What metric do you need and what must you measure to obtain it?" Examples of metrics are "23% reuse," "225 SLOC/LM," and "1 defect per KSLOC."

Table 2 shows how the GQM paradigm is applied to some management goals. Asking the questions in the second column yields the examples of metrics shown in the last column. These examples of metrics are basic measurements that should be made on each system product at each activity or measurement point in the development process. A systematic measurement program designed for a specific organization may require more data than is shown here, so this set of measurements is a representative set.

Table 2. Goal-Question-Metric Paradigm Applied to Management Goals

Goal	Question	Metric Category	Examples of Metrics
Manage and control the project	How much have we made? How much is left to be made (progress)?	Size	SLOC Boxes Procedures Units of output
	How much progress have we made?	Status	Earned value Amount of scheduled work actually done Percent of each activity complete
	How much effort has been expended?	Effort	Labor hours in enough detail to differentiate these activities: requirements, design, implementation, and test
	When will the product be completed?	Schedule	Calendar time (months, weeks) of activity completion
Manage and control the product	How good is the product?	Quality	Defects found Mean time to failure Mean time to repair
	How effectively does the product perform?	Performance	Technical performance measures specified by customer and management
Improve the process	How cost efficient is the process? What is our present performance?	Time and effort	Unit costs Time to complete
	How good is the process at creating products that meet requirements?	Technical performance measures and quality	Selected measures for technical performance and defects for quality
Manage the risks	What are the risks?	Risks	Probabilities of exceeding constraints or not meeting requirements

This page intentionally left blank.

4. THE SYSTEMATIC MEASUREMENT PROGRAM

Management begins by identifying the performance improvements that are most urgently needed and . . . sets about at once to achieve measurable performance in a short time.

R.H. Schaffer and H.A. Thomson, "Successful Change Programs Begin With Results"

4.1 SYSTEM MEASUREMENT

This section describes the principle activities or process for developing software-intensive systems. It describes the use of measurement to characterize those activities and their interconnections. It defines the principle subprocesses of developing hardware, software, and procedures and describes the system life cycle. Throughout, this section stresses the central role of measurement in managing the development of software-intensive systems.

This report's view of systems engineering, the system life cycle, and systems development management applies to DoD-sponsored development efforts. Commercial system development efforts generally have a different view of the life cycle and systems engineering, but it is possible to map from one view to the other and thus show that they are analogous. It should also be noted that the view of systems engineering presented in this report is wider in scope than that implied in MIL–STD–499B (Draft) (Department of Defense 1993b) and does not represent an interpretation of that draft standard.

4.1.1 The System Development Process

The system development process refers to the entire set of activities and their sequencing that takes place over the life of a software-intensive system (Department of Defense 1991b). A system's life commences with the identification of a need for the system (in the form of a mission statement) and ends with the retirement of the system. The specific life-cycle phases may vary in their relationships to each other and in their duration due to the nature of the system, the purpose, and the environment of operational use. Because a software-intensive system is viewed as a combination of the major subsystems of hardware, software, and procedures (including people), the system life cycle has analogous phases or activities in each of these major subsystems.

From the point of view of DoD, the system development process is often characterized as "the acquisition process" (Department of Defense 1991b). The acquisition process includes the activities of conceptualization, development, production, deployment, and system support as well as the activities of contracting, proposal evaluation, and awarding contracts.

This top level view of the acquisition process implies that, to find the activities of system development, there must be successive breakdowns of these phases through multiple levels into the separable activities of system development. And, because these levels and these activities have associated metrics, the activity-based representation of the system development process is actually a top-down approach to creation of a management information system for system development metrics. The acquisition process will be the life cycle discussed in this report.

The life cycle for software-intensive systems consists of the following phases:

- The Concept, Exploration, and Definition Phase (Phase 0) is triggered by an identification of a need. It consists of advanced program and top level system product planning activities.
- In the Demonstration and Validation Phase (Phase 1), critical design characteristics and expected capabilities are demonstrated, validated, and refined.
- The Engineering and Manufacturing Development Phase (Phase 2) translates the most promising design concept into a system that meets requirements.
- The Production and Deployment Phase (Phase 3) establishes the support base for production.
- The Operations and Support Phase (Phase 4) ensures that the fielded system continues to meet mission needs throughout its life cycle.

The system requirements analysis and the system *conceptual design activities precede the* hardware, software, and procedures development processes shown in Table 3. For software-intensive systems, these development processes often occur simultaneously (Department of Defense 1985, 1988). Table 3 shows how similar these activities are for development of hardware, software, and procedures.

Table 3. System Development Process Activities for Hardware, Software, and Procedures

Hardware	Software	Procedures
Analysis of requirements allocated to hardware	Analysis of requirements allocated to software	Analysis of requirements allocated to people, procedures, and logistics
Preliminary design	Preliminary design	Preliminary design
Detailed design	Detailed design	Detailed design
Fabrication and testing of individual hardware units (HWU)	Coding and testing of individual computer software units (CSU)	Preparation and testing of individual procedures
Integration and testing of HWUs into hardware component (HWC)	Integration and testing of CSUs into computer software components (CSC)	Integration and testing of procedures
Testing of hardware configuration items (HWCI)	Testing of computer software configuration items (CSCI)	Testing of procedural configuration items
Testing of complete system		

Appendix C discusses the systems engineering process, which is a subprocess of the system development process.

4.1.2 Measurement in the System Life Cycle

Measurement is associated with the system development process, product, and project and is necessary at every phase of the development life cycle. This planned and thorough system measurement is what defines a program of systematic measurement.

Figure 3 shows the concept exploration and definition (Acquisition Phase 0) and demonstration and validation (Acquisition Phase 1). It includes generation of the software, hardware, and procedures requirements; the design process; and the implementation and test efforts. Under many conditions, visibility is also required into the post-deployment support and enhancement effort, although it is not shown here. Measurements of size, cost, schedule, and quality are usually made at each numbered arrow. Figure 3 shows that measurement is an integral and continuous process throughout the development cycle and that the measurement process is shown by eight measurement points in the figure.

Figure 4 shows the details and the measurement points of the engineering and manufacturing development phase (Phase 2). The subsequent phases of production, deployment, operations, and support (Phases 3 and 4) are shown but not detailed in this report.

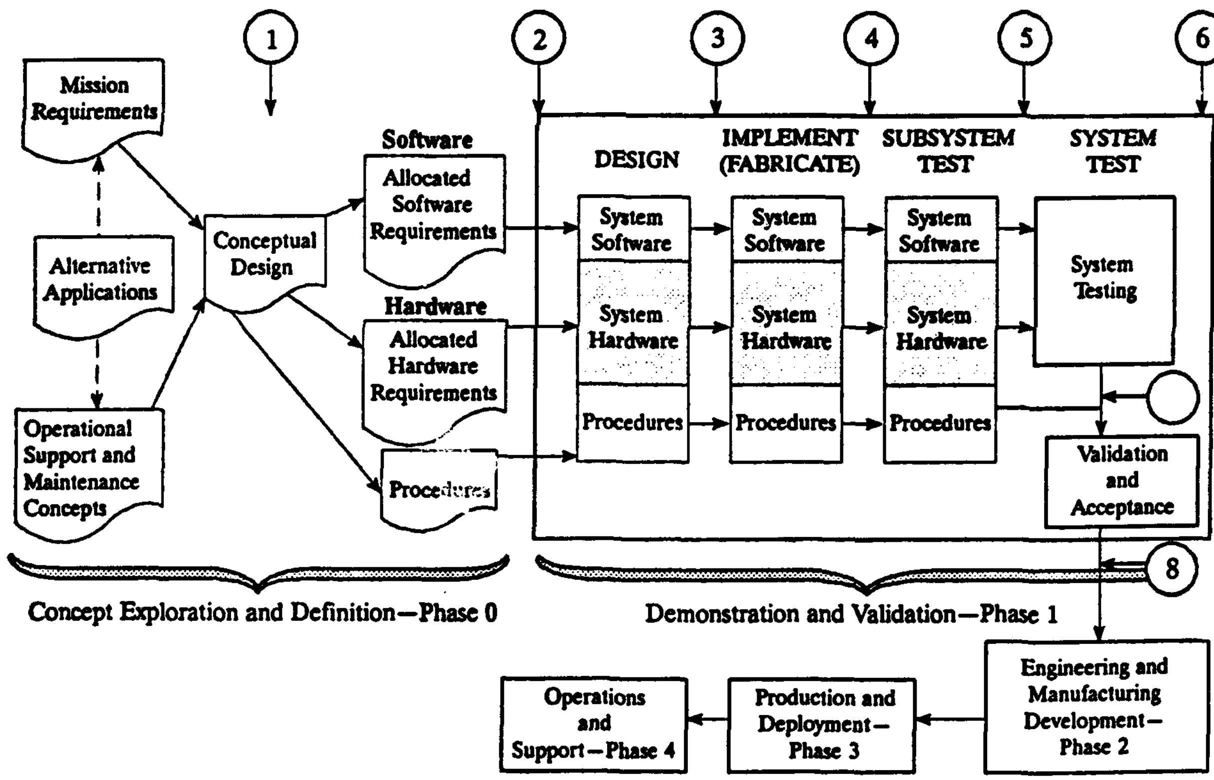

Figure 3. Acquisition Process Phases 0 and 1

Table 4 shows the typical metrics and measurements that can be collected at each measurement point in the acquisition life cycle. The decision of what measurement to collect depends on the project, the environment, and the needs of management. The GQM paradigm should be employed to select the "right" measurements, i.e., measurements that support project and management goals. If GQM is used, then neither too many or too few measurements will be collected. Table 4 should be viewed as a "menu" for guidance in the selection of measurements. Suggested metrics are not given in detail for acquisition Phases 3 and 4 because those phases are not discussed in this report.

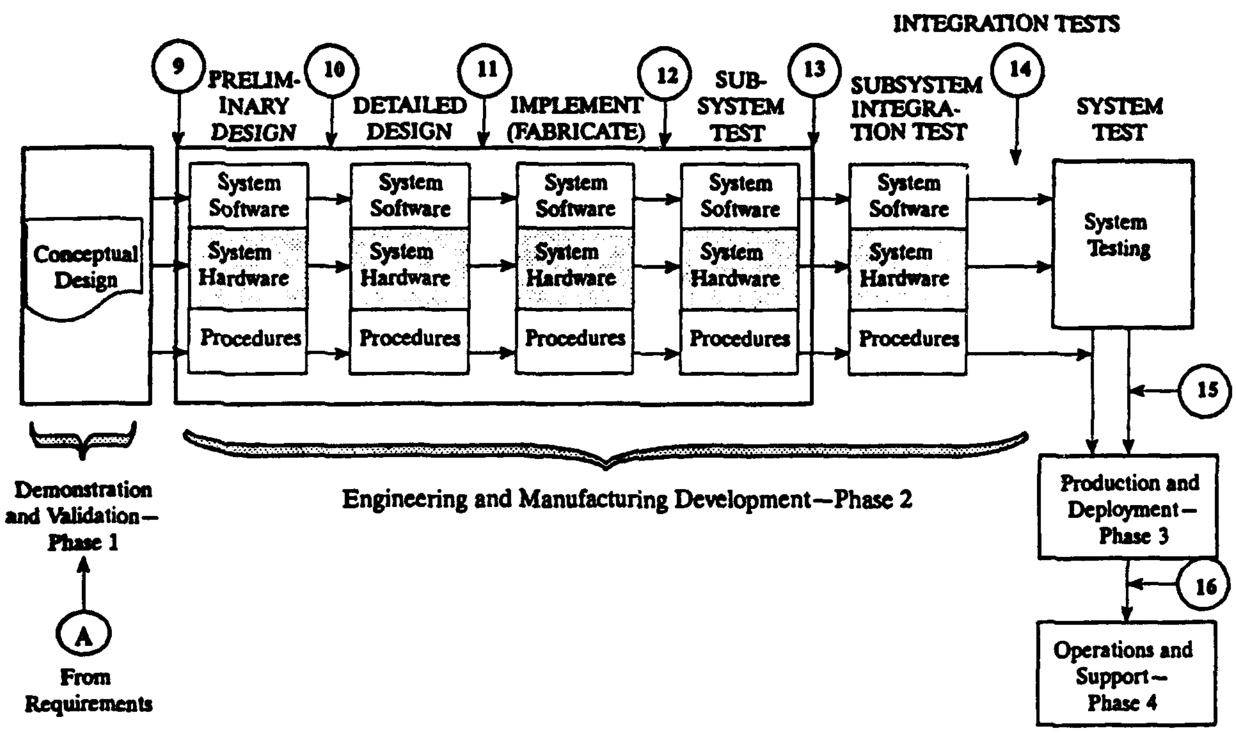

Figure 4. Acquisition Process Phase 2

4.2 SYSTEMATIC MEASUREMENT

Organizations have both a need and a responsibility to make action-oriented information available to the appropriate management levels in a timely and useable form. This need can be satisfied by a planned and organized program of quantitative information collection, organization, analysis, and dissemination to support management decision making. Such a program is called a systematic measurement program. Metrics data that can be collected is shown in Table 4.

4.3 GOALS OF SYSTEMATIC MEASUREMENT

The generic system measurement program discussed here is part of a larger program of project management and of continuous process improvement. Because the monitoring and control of ongoing development projects require the collection and analysis of metrics data on progress and status, it is necessary to articulate the organization's goals and to decide how the metrics program should support these goals.

Table 4. *Metrics and Measurements That Can Be Collected During the Life Cycle*

Measurement Point	Acquisition Phase	Metrics and Measurements
1	0	Number of mission requirements Number of alternative conceptual designs Cost and schedule estimates for each alternative Number of operations, support, and maintenance alternatives Required quality (defect densities)
2	1	Number of requirements (hardware, software, procedures, and logistics) Number of systems and subsystems Cost (effort) and schedule estimates Number of subsystems and components Estimated number of source statements—new and reused
3	1	Actual number of software design statements Number of hardware subsystems and components designed Number of procedures designed Cost (effort) and schedule of each design activity or task Budget and planned schedule for each design activity Number of errors (defects) found during preliminary and detailed design reviews for hardware, software, and procedures Number of documentation pages Action items—open and closed Staff positions authorized and filled Process standards invoked
4–6	1	Actual number of source statements written and units tested Number of subsystems and components created Technical performance measures on hardware and software Number of procedures created Number of defects discovered in code inspections Number and type of defects discovered in component and subsystem test Number and type of defects discovered during system integration test Status and earned value at planned points Actual cost (effort) and schedule for each implementation activity Budgets and planned schedule for each implementation activity Number of tests and test procedures Percent of tests passed Test problems (trouble reports) Number of documentation pages Action item status—open and closed
7	1	Technical performance measures—required performance levels versus actual from test Evaluation data—number of subsystems and components recycled for design and implementation
8	1	Number of design changes for Engineering and Manufacturing Development, Phase 2 Estimated costs and schedule of design changes
9–14	2	Similar to measurement points 2–7
15	3	Number of design changes for Production and Deployment, Phase 3 Estimated costs and schedule of design changes
16	4	Number of design changes for Operations and Support, Phase 4 Estimated costs and schedule of design changes

Some typical organizational goals, as they relate to technical direction and associated system development, are to:

- Effectively support program and project managers in the technical execution of each project, i.e., help management control the projects.
- Administer and support technical personnel.
- Improve the maturity of the development processes for software-intensive systems.
- Maintain the technology base.
- Promote technology transfer.

The systematic measurement program should support these goals by enabling management to:

- Monitor progress toward the attainment of:
  - Project (control) goals
  - Process improvement goals
- Anticipate problems.
- Initiate corrective actions.
- Provide increased visibility where necessary.

5. THE ADOPTION OF A PROGRAM OF SYSTEMATIC MEASUREMENT

By replacing large-scale amorphous improvement objectives with short-term, incremental projects that quickly yield tangible results, managers and employees can enjoy the psychological fruits of success.

R.H. Schaffer and H.A. Thomson, "Successful Change Programs Begin With Results"

This section shows the major steps necessary to adopt a program of systematic measurement. It presents arguments against the adoption of systematic measurements and counterarguments in favor. It discusses investment in systematic measurement and the returns on investment (ROI) that can be expected.

5.1 PROGRAM ADOPTION

5.1.1 Key Roles in the Adoption Process

There are some key roles that must be filled in the adoption of a systematic measurement program:

- The **sponsor**, a senior or middle manager, who has authority to make commitments and assign resources. While this person often does not participate directly in the program development, his leadership and support are essential.
- **Champions**, who act as advocates for the program by virtue of their technical credibility and influence. Ideally, champions are managers at the senior, middle, or program or project level.
- **Change agents**, management leaders who are empowered by sponsors and champions to plan and implement the program. Change agents are members of working groups that will benefit from the systematic measurement program. Influencing or persuading by their knowledge or experience, the operational knowledge of change agents is needed to schedule, monitor, control, and report the accomplishments of the systematic measurement program.
- **Targets**, projects or organizations that are selected for specific implementation of the measurement program.

In addition to the key roles listed above, there are some supporting roles that are very important:

- **Idea generators**, who contribute new ideas concerning appropriate measurement program ingredients
- **Idea exploiters**, who take new ideas and implement them in the form of pragmatic programs
- **Information gatekeepers**, who provide informed realities about systematic measurement

5.1.2 The Adoption Process

The major steps in the adoption of a program of systematic measurement are:

1. ***Perceive the need for and a commitment to a systematic measurement program.*** The sponsor, who sees the need for better management information and commits an organization to a measurement program, should be a senior manager, such as a site manager, general manager, or division vice president. It is at this level that the lack of timely and usable quantitative information to solve major project problems becomes apparent, and it is this level that has the authority to ensure understanding at all levels in the organization. While the idea of a systematic measurement program may come from lower level managers who will be able to demonstrate their lack of information for decision making, the need is typically recognized and the program instituted by an upper level sponsor who is responsible for several major development programs.

2. ***Identify a champion and assign organizational responsibility.*** A champion is essential. This role must be filled by a respected advocate for the program, whose technical credibility and influence are widely respected. The champion is assisted by a change agent, who is charged by the sponsor to plan and implement the program. The best change agents are members of working groups that will benefit from the systematic measurement program. A systematic measurement program should reside in an organization that has the resources, authority, and responsibility to make the program happen. Unless a very limited program is contemplated, it is not sufficient to give the responsibility for implementing a program of systematic measurement to a single individual, no matter how high his level or how experienced because there is usually too much to do.

   This step should also include the identification of:

   - Management leadership for the systematic measurement program
   - Information gatekeepers who are experts in systematic measurement and who can provide great help in implementing the program
   - Idea generators and idea exploiters who can implement those ideas to form a workable measurement program
   - Methods to ensure acceptance of the systematic measurement program by all development employees

3. ***Establish tangible objectives and meaningful measurement program activities.*** The change agent guides the planning of the program, including the creation of program objectives and the design of program activities. The planning takes the sponsor's goals for more effective information and defines expected results, needed resources, tasks, and organizations responsible for affecting the program.

4. ***Facilitate management buy-in at all levels for the systematic measurement program.*** The upper level manager who sponsors the measurement program must clearly communicate to all levels of management to inform them of his interest in the measurement program and to motivate their cooperation. They need to know the implementation team's goals, the responsibilities, the authority, and the interfaces with other organizations. Equally important, other managers need

to "buy-in" to the program. An important step in obtaining this buy-in is to work with affected managers to tailor the implementation so that most of their needs are met.

5. ***Implement the systematic measurement program on an appropriate target of opportunity.*** Plan and execute the planned incremental implementation. Data collection should begin at the initiation of each project undertaken by the organization and must be carefully planned. Initially, the implementation team is responsible for collecting the newly required data. Organizations and managers that are expected to contribute data must be assured that it is an opportunity for process and product improvement. These organizations cannot be expected to contribute data in response to a written request from the team; they tend to plead both limited resources and limited knowledge of the data requirements. The implementation team should be empowered by the program sponsor to collect and validate data from any and all appropriate areas of the organization.

6. ***Review the systematic measurement program.*** After completion of step 5, the implementation team should document the work conducted, results obtained, and lessons learned that may expedite implementation of the next program. The sponsor needs to be informed of the status and accomplishments of the measurement program. The implementation team should inform all levels of management of program results. These include more accurate estimates for proposals, improved project tracking and monitoring efforts, risk reduction efforts, and productivity studies. The implementation team should share the data in its experience database freely with all levels of management and with all technical groups while maintaining strict control over what data goes into it.

Additional guidelines for adopting measurement technology may be found in the Consortium's *Using New Technologies: A Technology Transfer Guidebook* (Software Productivity Consortium 1993b).

5.1.3 Evaluating the Measurement Team

The measurement function should periodically evaluate its own performance and report to upper level management its effect on:

- Supporting management with accessible, timely, and meaningful data and metrics, which usually means the establishment of an experience database
- Planning budgets that accurately reflect the estimated costs
- Planning schedules that realistically reflect the size and scope of the products
- Creating and managing with WBSs that reflect the project organizational structure at all levels
- Reducing cost risk and schedule risk
- Improving product quality through better control of the development process
- Improving the development process by producing better products at less cost
- Creating and refining system standards

5.1.4 Systematic Measurement Program Tasks

A systematic measurement program can be organized in many ways. Regardless of the organizational approach, the program should be tailored to the particular development organization. The systematic measurement program includes the following tasks:

- Collect, analyze, and disseminate the following systems development project data:
  - Process and product cost
  - Schedule (time and milestones)
  - Size (weight, functions, components, etc.)
  - Quality data
  - Technical performance measures
- Establish and maintain a measurement database.
- Document and quantitatively characterize (i.e., measure) the product development process. This task involves knowing all activities in the product development process together with their cost and schedule profiles. This task can be extended to all internal processes after the initial task is completed. If this information is communicated to the appropriate levels of project management, they will use it to improve the process. The SEPG should work with the measurement program to accomplish this task.
- Derive and adapt estimation algorithms that are compatible with the system measurement environment. Communicate these methods to the estimators and show them how to use them.
- Make estimates of cost, size, and schedule in all proposal situations. The estimates can be made by the measurement function in parallel with the proposing organization and differences between the estimates can be negotiated to ensure a correct estimate.
- Track and monitor ongoing system development projects by collecting data and applying measurement methodologies to estimate project status, anticipate problems, and predict project and product performance. This information may also be organized into a formal database.
- Measure process improvement incrementally and through time so that trends are visible.
- Measure quality performance, and use these measurements to improve the product.

5.2 THE ECONOMICS OF SYSTEMATIC MEASUREMENT ADOPTION

There is a cost for implementing a program of systematic measurement and for making it an ongoing function. An investment is made to establish the measurement function and its systematic procedures, and recurring costs are incurred for applying systematic measurement to projects. The cumulative benefits to be derived over time and over all projects will be greater than the cumulative costs after the initial investment is paid for by the benefits of systematic measurement; i.e., the ROI goes from negative to positive when the break-even point is reached. Figure 5 is a notional diagram that illustrates the economics and timing of systematic measurement adoption. Note that, after instituting a systematic measurement program, the organization has a new baseline and traverses a new path.

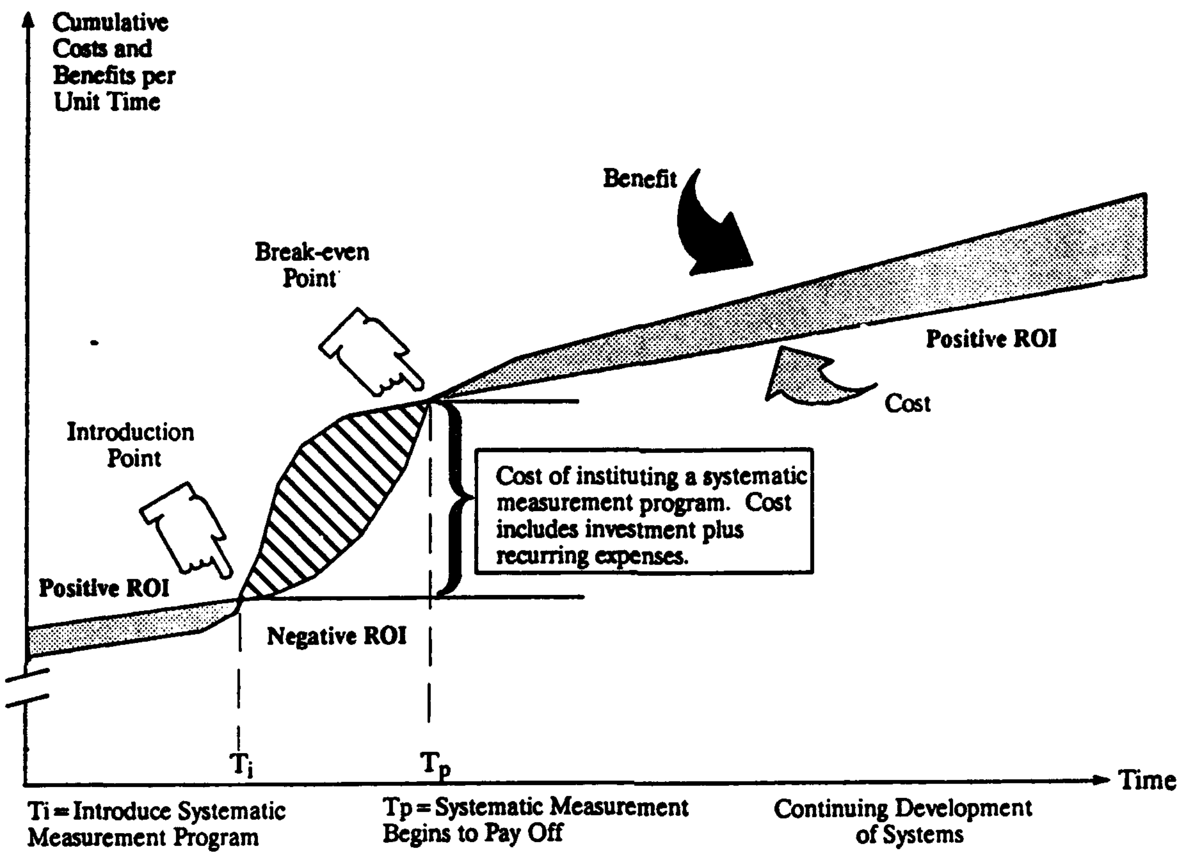

Figure 5. The Economics of Systematic Measurement Adoption

5.2.1 Investment Strategies

One possible way to "invest" in a program of systematic measurement is to fund the program from overhead money for some predetermined time, such as 1 year. During that year, the measurement program can derive and validate estimating algorithms and methodologies, set guidelines or internal standards and procedures, begin to collect data for a metrics or measurement database to support management, and initiate project tracking and monitoring. This overhead initial "investment" can be considered a one time startup investment

If the systematic measurement program is deemed successful at the end of the year, the recurring measurement activities can be funded with contract funds (the 2% to 3% of direct development costs mentioned previously) from that point on. This incremental method of funding will cover the recurring costs of measurement.

Another possible way to invest in a systematic measurement program is to make an investment in only those measurement activities that enhance the organization's knowledge base. Those activities occur over time and might include:

- Collecting data and updating the measurement experience database
- Deriving and refining measurement procedures and estimation algorithms
- Establishing and refining internal organization system standards

Because the above measurement activities go on more or less continuously over time, this method of investment is really an investment stream and not just an initial (one time sunk-cost) investment. Recurring costs of measurement, such as data collection and tracking, can be funded from contract funds.

A third investment strategy is not to make any initial investment but to fund a systematic measurement program incrementally from current contract funds. The payoff time for this strategy may be longer because the investment in knowledge and experience is made over a longer time and with less initial funding than the other two investment strategies.

5.2.2 Recurring Costs of a Systematic Measurement Program

As stated previously, experience shows that the recurring costs for a measurement or metrics program may be an amount equal to about 2% to 3% of the system development cost on average. This cost will vary with the organization's information-action maturity level (see Appendix B). At Maturity Levels 4 and 5, the costs may be as high as 4% of the system development budget. Continuing investment will be made and will vary with the extent of the systematic measurement program. When an organization budgets for a system development project, it should set aside, in the manner of a management reserve, an amount equal to the percentage of the direct labor system (hardware, software, logistics, and procedures) budget to support the measurement effort as applied to that project.

The system development projects should understand that, although measurement will cost the developers 2% to 3% (or perhaps 4%) of their budget, the future payoff to the project will more than pay for this expense through cost avoidance (see Figure 5 and Section 5.2.4). The measurement program budget should be a separate cost account in the financial reporting system.

The direct operating costs of a systematic measurement program derive from the following activities:

- Estimating size, cost, schedule, and quality for proposals and new projects
- Tracking cost, schedule, quality, and status of ongoing system development projects
- Reporting to the appropriate levels of management
- Statistical quality control and prediction reliability (where applicable)
- Training system development and management personnel in measurement

Because these activities will continue over time and over all projects, these costs are not one-time costs but are a stream of costs. These are project control or management costs that result in added value to the project and products, but they are not investments.

5.2.3 Return on Investment

The determination of ROI depends on how the benefits of the measurement program are stated. The benefits of measurement are savings from:

- Earlier identification of problems and earlier decision making
- On-time delivery
- Avoiding or reducing cost overruns
- Process improvement and increased productivity
- Avoiding costs of rework
- Avoiding borrowed funds (i.e., interest savings)
- Using resources in more productive endeavors that produce more profit
- Better estimates of cost, size, schedule, and quality
- Better product quality and trustworthiness

The total of these cost savings and avoidances is the total benefit derived from systematic measurement. Because both benefits and investment occur incrementally over time and over multiple projects, it might be better to use the present value of the benefits as they are expected to occur and the present value of investments are they are expected to occur to calculate the expected ROI of the systematic measurement program.

5.2.4 Examples of Investment Scenarios

This section provides two sample scenarios illustrating the ROI that can be realized from a systematic measurement program.

5.2.4.1 Scenario 1

A division vice president and general manager of a $500-million per year business decides to establish a systematic measurement program. Further, measurement function is established to provide data to manage the business better. The measurement function might be composed of a manager who is knowledgeable about system development, a system analyst who is a measurement practitioner and database specialist, and a systems engineer with system development and estimation experience. The decision is to invest only in establishing the systematic measurement procedures at the system level, the estimating algorithms and standards, and in structuring the measurement experience database. (The labor for the measurement function is to be charged to current contracts.) These costs are to be capitalized because they add to the knowledge base of the business. Assume that the postulated measurement function requires 4 months of effort to perform. The cost for these activities is 4 months for three persons at $20,000 per person month or $240,000.

In the next 2 years, two system development contracts worth $50 million each avoid a 10% cost and schedule overrun because the measurement function was able to provide advance warning of the problem. Total cost of the avoided overruns would be $10 million; if this were a cost plus fixed fee contract, the government would pay the contractor the extra $10 million but would pay no fee. If the overruns had not occurred, the system developer could have used that money on contract activities that would produce a fee and profit. Assuming an 8% allowed fee, there is $800,000 of fee and profit that was not lost because the overruns were prevented. Therefore, in 2 years, the ROI is ([$800,000/$240,000]-1)100 or 233%.

Cost avoidance and other returns after the second year will make the ROI much higher.

5.2.4.2 Scenario 2

The general manager decides to invest in the cost of starting the measurement group for 1 year with the goal of having the measurement function supported by contract funds after the year is over. The approximate cost of three technical staff for 12 months at $20,000 per month is $720,000. This is the investment.

In developing a mission-critical computer program of 500 KSLOC, the measurement function discovers that the software will not reach the required level of 0.5 defect/KSLOC at delivery. Instead the predicted defect density will be at the 1.8 defects/KSLOC level. At this level, the developer will be liable for the costs of much rework. Assume that management, with this advance warning, is able to change the development process by iterating through the design and code inspection activities to reduce the estimated defect density down to the specified 0.5 defect/KSLOC level.

A reduction of 1.3 defects/KSLOC in a 500-KSLOC program is 650 defects. Based on experience (Software Productivity Consortium 1992), it costs 0.6 person month per defect to fix errors after delivery of the software. This figure includes technical effort, configuration management, and support effort. Therefore, 650 defects at 0.6 person month per defect at $20,000 per person month gives a total savings of $7.8 million. This saving corresponds to an ROI of ([$7,800,000/$720,000]-1)100 or 983%.

These scenarios are realistic. They do happen. Systematic measurement does pay off.

5.2.5 Strategy for Incremental Adoption

The costs of developing a quantitative management system are controllable by implementing the program in several discrete blocks. Beginning with a minimum data set defined by the organization and collected for pilot projects (selection criteria may include the willingness of a customer to cost-share), its value to the organization can be assessed.

Major benefits of quantitative management come from reducing an organization's cost of poor quality and inadequate information. In U.S. industry, quality costs alone amount to a surprisingly large 20% of billings (Crosby 1980), which is **not identified in conventional management reports.** This large total does not stop with scrap. It includes rework, inspection labor, engineering changes, software correction labor, quality control labor, test labor, warranty costs, service costs (except regular maintenance), and other costs of doing things wrong the first time and then correcting them.

5.2.6 Incremental Blocks of Measurement Information

This section presents a possible strategy for data collection in incremental blocks. In the beginning state, only limited data is collected, according to the requirements of individual contracts. No visibility is available below the levels of CSCI/HWCI. With each increment of collected data, the initial highly aggregated levels are refined, providing increasingly greater visibility with each block. The purpose of this implementation strategy is to minimize the cost of collecting each incremental block of measurement data so that net benefits may be obtained within a few months after startup.

Figure 6 is a notational diagram that illustrates the cumulative value of incremental adoption of measurement. The plot shows the effects of both incremental data blocks and of implementation on additional projects and programs.

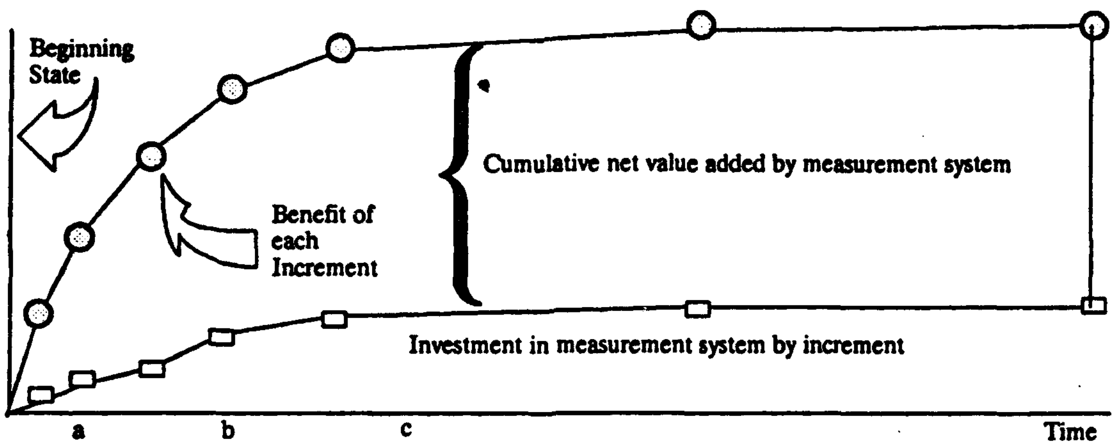

Figure 6. The Cumulative Value of Incremental Investment in Measurement

- ***Minimum Data Set.*** The first increment of data, consistently collected to the CSCI/HWCI level, adds visibility on size, staffing, schedule, cost, status of requirements changes, and technical performance measures. Data is collected monthly or weekly. The program can begin to plot profiles for actual or plan over time.

- ***Increment A.*** A second increment of data adds the first indicators of quality by collecting counts of defects or failures reported in test. This block also extends the detail collected by one level, making visible data on the components that comprise the CSCI/HWCI level. Data begins to be identified to the CSC and HWC level.

- ***Increment B.*** This increment extends identification another level to the units that make up the CSC and HWC components and to specified processes, such as formal design level inspections. The program can begin to plot rework ratios for CSCI/HWCI level.

- ***Increment C.*** This increment extends identification to product-line-specific measures and to more processes. At this level, analyses provide process control limits, such as productivity ranges and variances, for major activities and processes.

5.3 IMPEDIMENTS TO ADOPTION OF A SYSTEMATIC MEASUREMENT PROGRAM

5.3.1 Arguments Against Systematic Measurement

In some instances, management may be aware of the need for measurement and a measurement program but may be reluctant to institute a systematic measurement program because of the following common perceptions:

1. Systematic measurement costs too much; too much investment is required, and the return is too low.

2. We have all the data that we need to support special studies for the general manager.

3. We have the ability to measure when and if we need to.

4. Our estimates are based on standard industry methods, and our budgeting and planning are good enough.

5. If we collect all this data, the government (or the prime contractor) will want to see it and may take it away and use it to harm our organization.

6. We do not believe in quantitative systems management, and we do not have a program of systematic measurement.

5.3.2 Counterarguments Favoring Systematic Measurement

1. Some actual experience with systematic measurement suggests that recurring costs of 2% to 3% of project direct costs are adequate for data collection and analysis and for project tracking and monitoring. This small price buys real help in meeting project goals, increased project control through better budgeting, problem anticipation and amelioration, in risk reduction, and incremental process improvement.

2. Many organizations have data in many forms scattered over the whole organization, but that data is not organized or available and is not accessible on a timely basis. The general manager is not the only level of management that needs and consumes measurement data. All levels of management need measurement data in meaningful form. Lower levels of management may need information in different forms, i.e., more detailed quantitative technical information than the general manager, but all levels need the information that the measurement function can provide.

3. Many organizations have the ability to measure their performance, but they only do it when a problem is apparent. At that point, appropriate information may not be available, if it exists at all, in time to solve the problem. System measurement, if practiced in a systematic manner, ensures that information is available at all times for all projects over all levels of management for problem solving and decision making.

4. To be good enough, estimates, estimating algorithms, metrics, and experience data need to be tailored to the organization's unique environment and processes. Industry standard estimating algorithms, while useful, must have their parameter values calibrated to reflect the organization's unique environment; otherwise, they produce estimates that are not meaningful or reliable in that environment. Experience shows that controllability of system development projects decreases when budgets and the budgeting process bear little relation to the operating environment.

5. The government has access to all government contract data as individual data items and can compel, if necessary, a contractor to give it access to those data items. It is not clear if the government can compel a contractor to give it access to a central measurement database, i.e., an organized collection of selected data items that are, in effect, management data not collected as a part of the contract. The measurement database will contain information from past projects as well as ongoing projects. After a contract is satisfactorily completed, it is unlikely that the old data will be requested. Because this database will prove vital to the management of the business, it should be kept under a reasonable level of security. Many DoD contractor organizations collect only the data required to be provided to the government. Often, this set of data items does not include data that is needed by the development organization for its own use in project control and process improvement.

6. If you cannot measure your business, you cannot manage your business successfully for long. You cannot manage your business without information. Reliable information about your business requires measurements.

5.4 SUMMARY

In summary, adoption of a systematic measurement program makes good business sense:

- External forces are pushing you to measure, and internal forces are pulling you to measure.
- If you can't measure it, you can't manage it.
- A program of systematic measurements results in timely and meaningful information for management decision making.
- The use of GQM results in the efficient selection of metrics by ensuring that only the right things are measured.
- It makes good economic sense to invest in measurement.
- Measurement can be adopted incrementally, using incremental investments and getting incremental and timely returns.
- There are successful measurement programs in existence that continue to pay off for organizations today.

This page intentionally left blank.

APPENDIX A. THE STATE OF THE PRACTICE OF SYSTEMATIC MEASUREMENT

Appendix A presents two perspectives on the current state of the practice of systematic measurement. The first is a survey on current system measurement programs and practices as perceived by systems engineers attending the System Engineering Workshop organized by the Consortium (Software Productivity Consortium 1993a). The second perspective is drawn from the Executive Round Table also organized by the Consortium.

A.1 A SURVEY OF SYSTEM MEASUREMENT PROGRAMS AND PRACTICES

The survey was conducted with a sample of lead systems engineers and systems engineering managers, mostly from Consortium member companies, who attended the System Engineering Workshop held at the Consortium in June 1993. They responded to a survey on the system measurement experience and practices in their organizations. The questionnaire was brief so that the respondents could complete it in a short time. The responses are listed below in a question–response–interpretation format.

A.1.1 System Measurement Programs

1. Question: Is there a formal system measurement program in your organization?

   Response: Yes—44%, No—56%.

   Interpretation: A yes response includes the alternatives of a centralized formal program and a noncentralized or fractionalized program. A no response includes the alternatives of an informal program and no system measurement program. Apparently, most organizations do not perform formal (systematic) measurement.

2. Question: Does your organization have established standards for system measurement?

   Response: Yes—60%, No—40%.

   Interpretation: "Established standards" was interpreted by the respondents as either an established process with well-known activities or a documented standard process that included standard financial reporting and standardized technical performance measures. Because most organizations measure to at least this extent, the majority of the responses were affirmative. The large number of no responses may indicate that the existence of standards is unknown to the respondents.

3. Question: What organization is responsible for system measurement?

Response:

Table 5. Organizations Responsible for System Measurement

Responsible Organization	Percent of Total Responses
Project or program office	29
Product development	24
Functional organizations such as finance, controller, or manufacturing	12
Planning	12
Systems engineering	6
No single organization	17

Interpretation: The variety of responses reflects a noncentralized or informal system measurement practice. It is apparent that systems engineers are often used for system measurement activities by organizations other than systems engineering because of their initial involvement with specification of the system and their subsequent involvement with measuring product performance.

4. Question: Do you measure just at the *system* (product) level or also at the subsystem level?

Response: System only—7%, system and subsystem—80%, none—13%.

Interpretation: System and major subsystem level measurement is prevalent although not always centralized or tightly organized.

A.1.2 System Measurement and Metrics

5. Question: What do you measure at the system level?

Response:

Table 6. System Characteristics Measured

Measurable or Metric	Percent
Cost or effort to date	94
Adherence to budgets	100
Estimates to complete and estimates at completion (cost)	94
Product size and weight	87
Subsystem and component count	73
Financial and technical status	100
Adherence to schedule	100

Interpretation: A large variety of metrics are collected and analyzed by system measurement practitioners.

A.1.3 System Measurement Estimation Models

6. Question: What estimation models are used before or at project initiation?

Response:

Table 7. System Estimation Methods Used

Measurable or Metric	Percent Using Estimation Models and Methods		
	Experience Data	COCOMO or SASET or PRICE (S,H,L)	Rough Estimates Only
Cost (effort) and product size and development schedule	65	55	13
Product quality	0	0	0

Interpretation: Respondent organizations use a variety of models and methods to estimate cost, size, and schedule. Experience data is available (in local databases, as in Question) and is widely used. Product quality is either not estimated or not tracked at the system level.

A.1.4 System Measurement Schedules

7. Question: When do you measure?

Response:

Table 8. System Measurement Points

Measurement Points	Percent of Yes Responses
Preselected times	80
Regular time intervals	77
Milestones	100
Program-specific times	85

Interpretation: Respondents' organizations plan measurement activities for a variety of points and events in time. The selection of measurement points depends on a mixture of standard practice and program requirements.

A.1.5 System Measurement and Metrics Utilization

8. Question: What do you do with the measurements and metrics?

Response:

Table 9. System Measurements and Metrics Utilization

Utilization	Percent of Yes Responses
Determine financial and technical status	100
Determine product viability and adherence to specifications	93
Predict cost and status	100
Evaluate and update plans and specifications	100
Identify or anticipate problems	94

Interpretation: Wide use is made of a variety of measurement data.

9. Question: What special measurements or other information in areas of concern are obtained for the management levels of President/General Manager, Vice President/Director, and Engineering Manager?

Response: Because of the similarity of the responses, these three management levels were combined into one table. The numbers total to 100%:

Table 10. Areas of Senior Systems Management Concern

Areas of Management Concern and Information Utilization	Percent Indicating That Area of Concern Was the Most Important to Management
Cost	25
Schedule	18
Technical and financial details	15
Project status	13
Size and weight	12
Risk (cost, schedule, performance)	9
System performance	5
Resource utilization	3

Interpretation: The upper management levels are most concerned with the areas of cost, schedule, technical performance measures, and status (such as earned value).

10. Question: What actions are taken by management as a result of the system measurement data collection and analysis?

Response: Reports, schedule and resource adjustments, personnel changes, overtime assignments, priority changes, corrective action, identification of fallback positions, and scope of work changes.

Interpretation: These are the expected reactions.

A.1.6 System Measurement Database

11. Question: Do you maintain a database of system measurements?

    Response: Yes—76%, No—24%.

    Interpretation: The yes responses apply to localized databases or organization-specific databases, e.g., the schedule database, the size and weight database, or the financial cost database. There do not seem to be any instances in this sample of centralized and integrated system measurement databases. Because the responses to Question 6 indicated that 65% use experience data for estimation, the databases referred to either are not used to the fullest extent or there are sources of experience data other than databases.

A.1.7 Survey Summary

The survey makes the following main points:

- Formal (i.e., systematic or planned) system measurement programs are rare if they exist at all.
- Measurement is common at both system and subsystem levels.
- Measurements are made mainly at the times and the milestones specified contractually.
- Measures monitored are primarily financial (cost and earned value), technical performance and status, and adherence to schedules.
- The local project database is the most common practice.
- The measures support a wide variety of management actions.

The responses to the survey indicate a variety of experiences relative to systems engineering measurement. Many of the respondents are not directly involved with measurement programs. Further, they may not be able to provide a full management perspective on the need for systematic measurements in their organizations or the precise manner and models to process the data that is needed. They were not asked about the usefulness of the data obtained from the measurements, and they were not asked about systems management metrics.

A.2 THE EXECUTIVE ROUND TABLE

The theme of the Executive Round Table held at the Consortium in October 1993 was "Management Indicators for Senior Executives." There were 36 senior executives from the Consortium's member companies with an interest in and an involvement with software measurement and metrics who participated. Four of the attendees gave presentations about applying management indicator metrics to the management of large software-intensive system development projects in the aerospace industry. These presentations were followed by a panel of speakers who answered questions and comments from the attendees. This section is a very brief summary of the presentations and the panel comments.

Software metrics have been and are being used widely by management at the senior executive level, mostly on DoD-sponsored development projects. Metrics are crucial in managing large design and development programs. There are challenges to those developing and applying metrics in their management activities. There is a need for rigor in the development and application of metrics in management.

Senior executives with responsibility for one or more large development projects usually want answers to the question, "Where are we (in the project)?" Many software development projects are tracked and monitored quite closely because experience has shown that such monitoring is necessary for discovering problems and ameliorating their effects. There is a general realization that this need for project tracking and status information can be satisfied by the class of metrics called management indicators. Such indicator metrics include not just cost and schedule metrics but also such indicators as software size growth, requirements stability, amount and impacts of software reuse, phase-related earned value, technical performance measures, and quality (phase-related defects per KSLOC). The goal should always be to measure the right things for each project and to report them, preferably with a metrics standard report.

Operating guidelines for managing large software development projects are to start with resources in place and with basic information under control: have good people, encourage open communications, use known processes and standard practices, and have available the right tools. Management should know the development process activities and tasks and map that process into the management indicators as shown in the metrics plan. The right tools include action item and trouble report control systems, software configuration management (CM) and a CM control board, a method to measure earned value (not just cost to date), and a central metrics database.

Controlling the project is much more than costing the project, and every element of control should be connected to at least one metric. The best practices available should be employed. These practices include participatory risk identification and quantification, concurrent engineering, internal design reviews, design to cost, alliances with subcontractors, and prototyping. Prototyping is especially useful for learning about the system to be developed and its problems.

There are open issues in the management of software development and software-intensive systems. The current development model, i.e., the waterfall model, is thought by some to be the wrong model for system development. The spiral model may be better, but few have applied it. There is some question about vendor and commercial off-the-shelf software (COTS) stability. The concurrent development concept is yet to be proved. Software technical staff needs and wants more training. Software reuse needs to be encouraged through proven economics benefits and management commitment. But progress has been made in many areas, such as languages (Ada), increasing standardization, and the emergence of open systems. Management indicators can contribute to an understanding of these issues.

The senior manager for developing software-intensive systems should take the following actions to ensure success in project management: control and evolve the software development model to meet new system requirements and environments; support reuse; insist on COTS wherever possible; commit to process improvement; use process assessment to spur improvement; encourage the development of new processes, such as those performed by the Consortium; and know the true costs of development, both direct and indirect. Successful system development follows from having a well-defined process, using metrics based on experience to measure performance and to improve the process, using quality (defects) data to identify problems and to change the process, and thoroughly training the software developers.

Future challenges for the senior executive with the responsibility for developing large software-intensive systems include the increasing necessity to think and plan with a domain focus, to be willing and able to change the process with changing technology, and to develop and apply integrated tools and methods.

The questions and comments of the senior executive attendees centered mainly on the use of metrics in systems engineering. Attendees wanted to know if metrics should be used in system design and what metrics were available, if systems engineers should design for understanding, how management indicators could contribute, and if the systems engineering process was known. The panel responded that there are design metrics available and that the Consortium has been active in this area. The conclusion of the panel was that much more work needs to be done in defining the systems engineering process and in measuring the activities of the process. The panel also observed that general-purpose estimation and prediction tools have been inadequate for special-purpose systems and that more work needs to be done in evaluating and improving such tools.

This page intentionally left blank.

APPENDIX B. THE INFORMATION-ACTION MODEL

Appendix B presents an information-action model for management to judge the level of meaningful metrics information available in their own organization and the level of management actions resulting from the information. This model may be used to determine the level of information maturity and the next stage of development in improving the information-action level of their organization.

B.1 MANAGEMENT ACTION AND THE INFORMATION-ACTION MODEL

How well can an information system support management actions? How can measurement systems do a more effective job of supporting managers' information needs? This appendix answers these questions by showing the relationship between information and action, and it relates management actions to the different kinds of information required at varying management levels. In this appendix, information is defined as processed measurement data that is of importance for decision making.

The higher levels of information usefulness should be examined first, where those responsible for taking action have the information they need both to decide on appropriate actions and to compare the results obtained with those expected. Such information:

- Is timely, permitting management to evaluate and take appropriate actions to avert problems and take advantage of opportunities as contrasted with only reacting to problems after they occur
- Is preprocessed, reducing large volumes of available data to much smaller amounts of information
- Compares performance to expectations
- Projects recent trends into the near future to help anticipate problems
- Identifies trends and highlights out-of-control conditions

These characteristics are those of good quantitative management information, on which management can base high-quality decisions.

B.2 INFORMATION AND ACTION

To contribute value, a measurement system must provide a variety of metrics, planned and organized with respect to both breadth and level of detail, to meet the time and complexity requirements of various managers' tasks. The purpose of any measurement system is to provide information to help managers identify present and potential problems and on which managers can base appropriate actions.

The senior manager needs visibility into process, product, and market trends for the broad programs in his responsibility. Lower level managers typically need more detailed data, often about details of individual parts of a project, as well as to meet contractual requirements.

The relationship between information characteristics and action in Table 11 follows the pattern established in a five-level Quality Management Maturity Grid (Crosby 1980). The scale relates the effectiveness of managerial actions, in terms of managers' ability to make decisions that affect their business positively, to the availability and quality of measurement information. The higher the level, the more meaningful is the information, the greater the predictability of events, and the greater the managers' ability to understand and influence outcomes in desirable directions.

Table 11. Information Characteristics and Action Scale

Stage	Information Characteristics	Action Characteristics	Predictability of Outcomes
V Certainty	Continuous	Opportunities are foreseeable.	Very high
IV Wisdom	Early	Mitigating action can be taken.	High
III Enlightenment	Timely	Minimum action is possible.	Modest
II Awakening	Late	Some action may be possible.	Occasionally
I Uncertainty	None	Problems are not foreseen; no action is possible.	None

B.3 FIVE STAGES OF INFORMATION USEFULNESS

The management actions taken depend on the level of the organization relative to measurement. At Stage I in Table 11, with no information or metrics available, problems are often not foreseen and managers cannot act to avoid them. At Stage II, when an organization has awakened to the need for consistent measures, some information may be available. Collection and analysis time is lengthy enough that the information is typically late relative to the time that it would have been useful to the decision-maker. Still, the profiles of actual versus trend data for a few data elements permit some problems to be foreseen and some corrective action to be taken. At the enlightenment stage, Stage III, timely information is typically available and problems can often be corrected promptly. At Stage IV, with the availability of forecasts and trends, managers can anticipate problems and avoid or mitigate them. Finally, at Stage V, both opportunities and problems can be predicted. Actions can be taken both to avoid problems and to take advantage of opportunities.

B.3.1 Management Measurement Actions

Table 12 shows Sage's four perspectives or management styles from which measurement-based action can be approached (Sage 1993). "... All of these perspectives ... are needed. Inactive and reactive measurements are associated with organizations that have a low level of process maturity. As one moves to higher levels of process maturity, the lower level forms ... become less and less used" (Sage 1993).

At Stage I, there is no comprehension of measurement as a management tool; the lack of data is seen as normal within the organization. At Stage II, it is recognized that measurement may be of value in diagnosing problems after they have occurred, but insufficient resources are available for a program of systematic measurement. At Stage III, the value of measurement is recognized and the beginnings of a program are seen. At Stage IV, systematic measurement is institutionalized. At Stage V, systematic measurement is an integral part of the management process, providing full support for attainment of business imperatives. All levels of management contribute to systematic measurement and benefit from it. Problems in system development are anticipated and corrective action taken using timely measurement information.

Table 12. Stages of Measurement in Action

Perspective	Stage	Measurement Characteristics in Action
Proactive	IV, V	Proactive measurements are designed to predict the potential for errors and help synthesize an appropriate life-cycle process that is sufficiently mature that the potential for errors is minimized.
Interactive	III	Measures are made as a product moves through various phases of the life-cycle process to detect problems as soon as they occur, diagnose their causes, and correct the difficulty through recycling, feedback, and retrofit to the process itself.
Reactive	II	Organization performs outcome assessment and, after it has detected a problem or failure, diagnoses the cause of the problem and gets rid of the symptoms that produce the problem.
Inactive	I	Organization does not measure at all, except in an intuitive and qualitative manner.

B.3.2 Information Characteristics by Measurement Usefulness Level

Table 13 shows examples of measurement information at each level shown in Tables 11 and 12. It gives examples of five characteristics of management information: timeliness; prediction; amount of detail preprocessed, summarized, or analyzed statistically; comparison with plan; and exception orientation. It should be noted that, in practice, management must define and endorse these measures.

Table 13. Typical Characteristics of Management Information

Stage	Timeliness	Predictive View Is Like	Extent of Preprocessing	Comparative	Exception Oriented
V	Vital information is available in near real time.	Weather radar	Essential elements of information, with trends, projections, and control limits	Consistent across product lines for all appropriate items	Identifies exceptions outside computed control limits
IV	Key data is available twice weekly.	Looking ahead, through light haze	Routinely available for key trends	For all necessary WBS items	Shows exceptions outside pre-specified ranges
III	Key data is available weekly.	Looking ahead, through heavy fog	Available with detailed request	For contractually required WBS items	By inspection of actual and plan ratios
II	Key data is available twice monthly.	Looking out of side windows	Totals provided for a few key items	Data at top level of project WBS	None
I	Key data is not available.	Rear view mirror	Must look through all data files	No data at WBS detail level	None

At the awakening level, Stage II, batch-processed information is available perhaps twice monthly and for only a few key items at the top WBS level. Information is of limited quality and timeliness. Thus, management actions still are limited to mitigating the adverse effects of events after they occur.

At the enlightenment level, Stage III, awareness of the needs makes information available weekly for contractually required WBS items. With some processing delay, details are available on request for items identified by report users.

At higher levels of measurement utility, those responsible for taking action need increasingly quantified information. Progressing from simple ratios of actual versus plan, the data becomes information as it evolves to include trends, projections, and control limits that help identify exceptions.

At the wisdom level, Stage IV, management information is available frequently for all WBS items down to a detailed level. When trends and ratios fall outside prespecified control limits, information is highlighted and may be shown on exception reports.

At the certainty level, Stage V, information is:

- Timely, permitting management to evaluate and take appropriate actions to avert problems and take advantage of opportunities.
- Forward looking, projecting recent trends and ratios into the near future.
- Predigested, reducing large volumes of available data to bite-sized amounts of trend and ratio information for easy evaluation.
- Comparative, showing actual attainments relative to expected actual and plan for all WBS items appropriate for management action.
- Routinely adjusting moving-average process control limits and highlighting situations in which actual and plan ratios differ significantly from current limits.

In summary, the information-action model shows how effective management action relies on the availability and quality of information.

APPENDIX C. SYSTEMS ENGINEERING

C.1 THE ROLE OF SYSTEMS ENGINEERING

Systems engineering is the discipline concerned with technical management of the processes used to develop software-intensive systems. It includes assessing the degree to which system requirements have been satisfied. It encompasses management of the technical aspects of the development of hardware, software, procedures, algorithms and techniques, integrated logistics, and quality assessment. Systems engineering adds value by increasing the level of confidence in performance of work in these areas through the imposition of common integration and partitioning procedures, methods for developing requirements, risk and other evaluation techniques, and engineering standards.

The role of systems engineering is shown graphically by Figure 7. In general, the systems engineering function defines system requirements and develops the top-level (advanced) system design using available technology from the stated mission and objectives of the system. At the other end of the development process, when the system and its components have been developed, systems engineering integrates and tests the system and validates that the system meets the requirements as stated at the front end. During system development, systems engineering manages the technology, assesses risk, and verifies that the development procedures will produce a system that meets requirements for all of the development areas: hardware, software, procedures, and logistics and support. The systems engineering management of technology concurrent with the functional area management of development is shown by the cross-hatched areas.

In general, a systems engineering methodology may be thought of as a management technology that consists of these elements (Sage 1992):

- A set of activities to be accomplished
- A set of methods and technologies
- A set of relations among the activities, methods, and people
- The environment in which each of these is embedded

To organize these activities, methodologies, and people, systems engineering makes its contribution throughout the system life cycle. This may be considered a succession of phases (Sage 1992). These phases are:

1. Requirements and specifications identification
2. Preliminary conceptual design

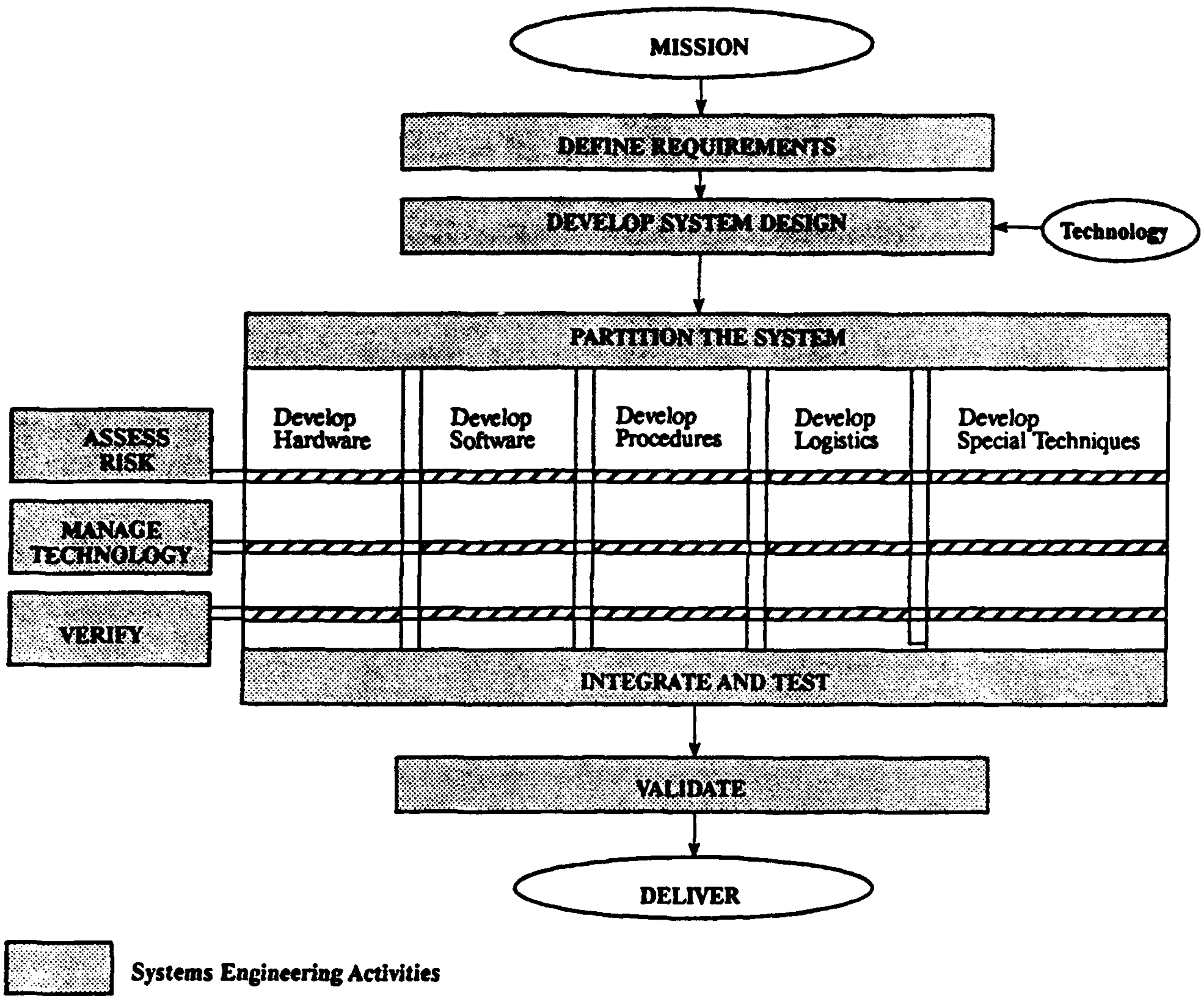

Figure 7. System Development Systems Engineering

3. Logical design and system architecture specification
4. Detailed design, production, and testing
5. Operational implementation
6. aluation and modification
7. Operational development

Systems engineering contributes to the success of program development in terms of cost, schedule, and technical performance.

A critical factor in the success of a system product development project is organization. Development people must be organized, the project must be organized, and the system itself must be organized. Systems engineering has a key role in all of these areas. Systems engineering is not the only contributor to system development; functional areas, such as product development, finance, management, research, marketing, etc., also contribute. But systems engineering is responsible, in most cases, for the critical system definition tasks of translating mission requirements to design requirements at successive lower levels, for assigning these design objectives to functional organizations, and for evaluating the results. This is the key role, both organizationally and technically.

C.2 THE SYSTEMS ENGINEERING PROCESS

Figure 8 shows a general model (Sage 1992) of the systems engineering process in terms of general tasks to be accomplished by systems engineering in a defined environment.

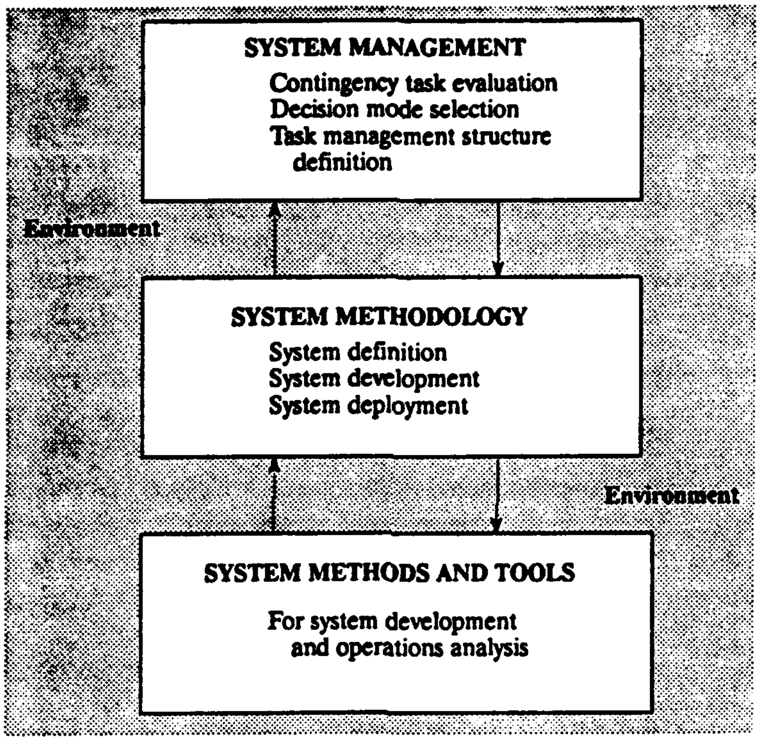

Figure 8. General Model of the Systems Engineering Process

Figure 9 shows a three-dimensional view of the system development process in which the development and application of hardware, software, and procedures (including logistics) are parallel planes, each with its own development process. The figure clearly shows the key role played by systems engineering in system development. These development processes do not proceed independently because there are many points of technical communication and information interchange, all managed by systems engineering. For example, systems engineering may approve a change in the design of the computer-based system hardware that affects both the software development and the logistics support subsystem.

Figures 7 and 9 illustrate the fact that systems engineering is a management technology; i.e., systems engineering involves the interaction of engineering science, the development organization, and the application environment (Sage 1992). The interactions among these three elements is in the form of information, and some of this information will be measurement information or metrics for the system. Qualification of system characteristics, i.e., system measurement, is a necessary part of system development and the systems engineering process.

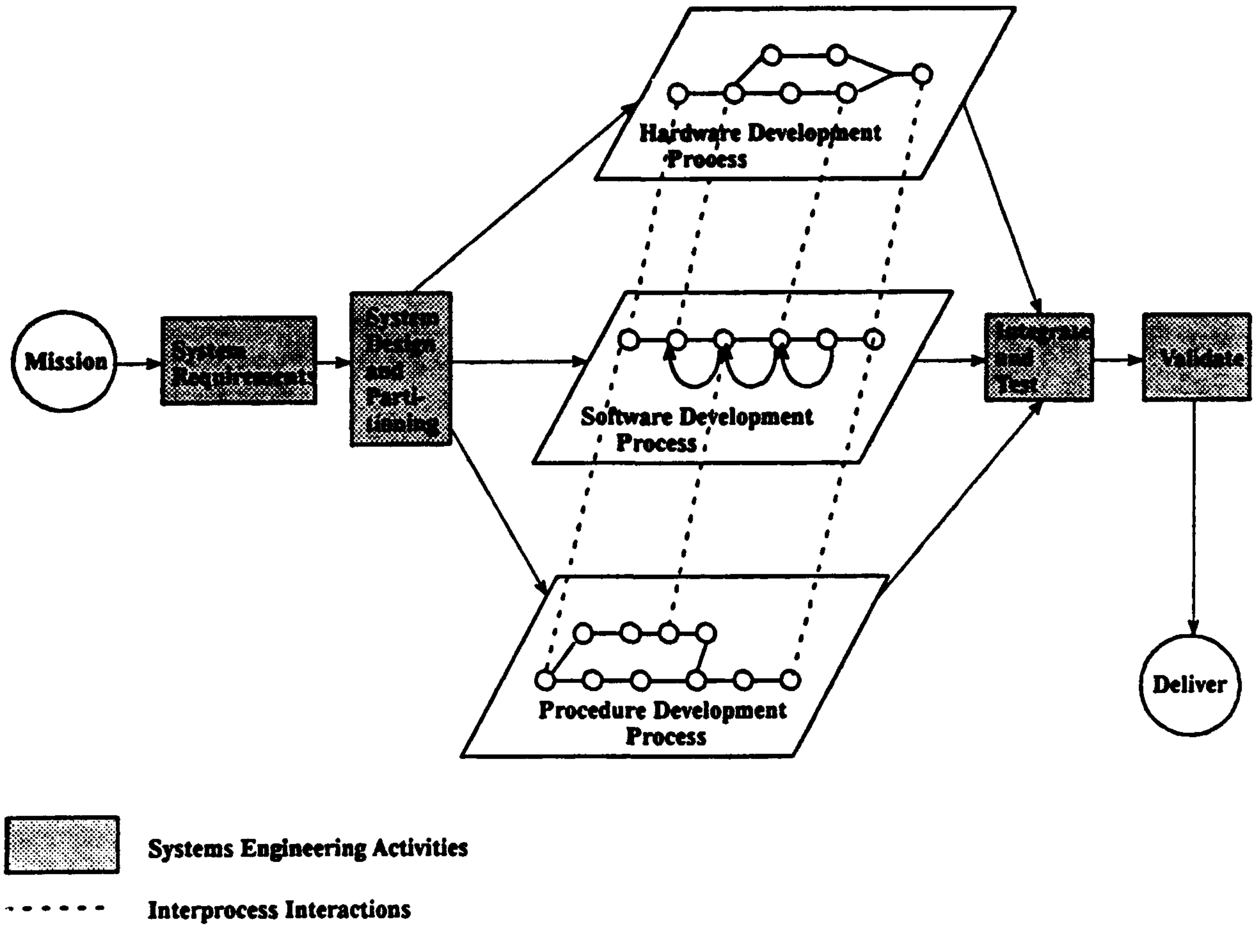

Figure 9. System Development Process With Technical Interchange Points

Sage (1992) defines a multistage systems engineering life cycle consisting of 22 phases. They are:

1. System definition phases

- Perception of need
- Requirements definition
- Draft request for proposal (RFP)
- Comments on RFP
- Final RFP and statement of work
- Proposal development
- Source selection

2. System design and development phases

- Development of refined conceptual architecture
- Partitioning of the system into subsystems
- Subsystem level specifications and test requirements development
- Development of components
- Integration of subsystems
- Integration of the overall system
- Development of user training and aiding supports

3. System implementation and maintenance phases

- Operational implementation or fielding of the system
- Final acceptance testing
- Operational test and evaluation
- Final system acceptance
- Identification of system change requirements
- Bid on system changes or prenegotiated maintenance support
- System maintenance change development
- Maintenance testing by support contractor

Systems engineering is involved in all of these phases and activities in the sense of managing the technology; verifying that requirements are met for hardware, software, and procedures (including system support); and assessing risk. The systems engineering process is a continual cycle of formulation, analysis, and interpretation through all of these 22 phases. There is much feedback to previous phases to modify the system design or to correct errors.

Much of the systems engineering process is devoted to front-end effort of system definition in the sense of translating requirements to architecture (Department of Defense 1992a; Blanchard and Fabrycky 1990; Blanchard 1991). These front-end activities can be viewed as a process consisting of requirements analysis, functional analysis and allocation, synthesis and system optimization, and system assessment and control. Together, these activities produce an architecture with appropriate specifications for the system in a defined environment. This process is shown graphically in Figure 10. It should be remembered that these front-end process activities are not the only activities of systems engineering throughout the system life cycle. Systems engineering is involved with all system life-cycle activities.

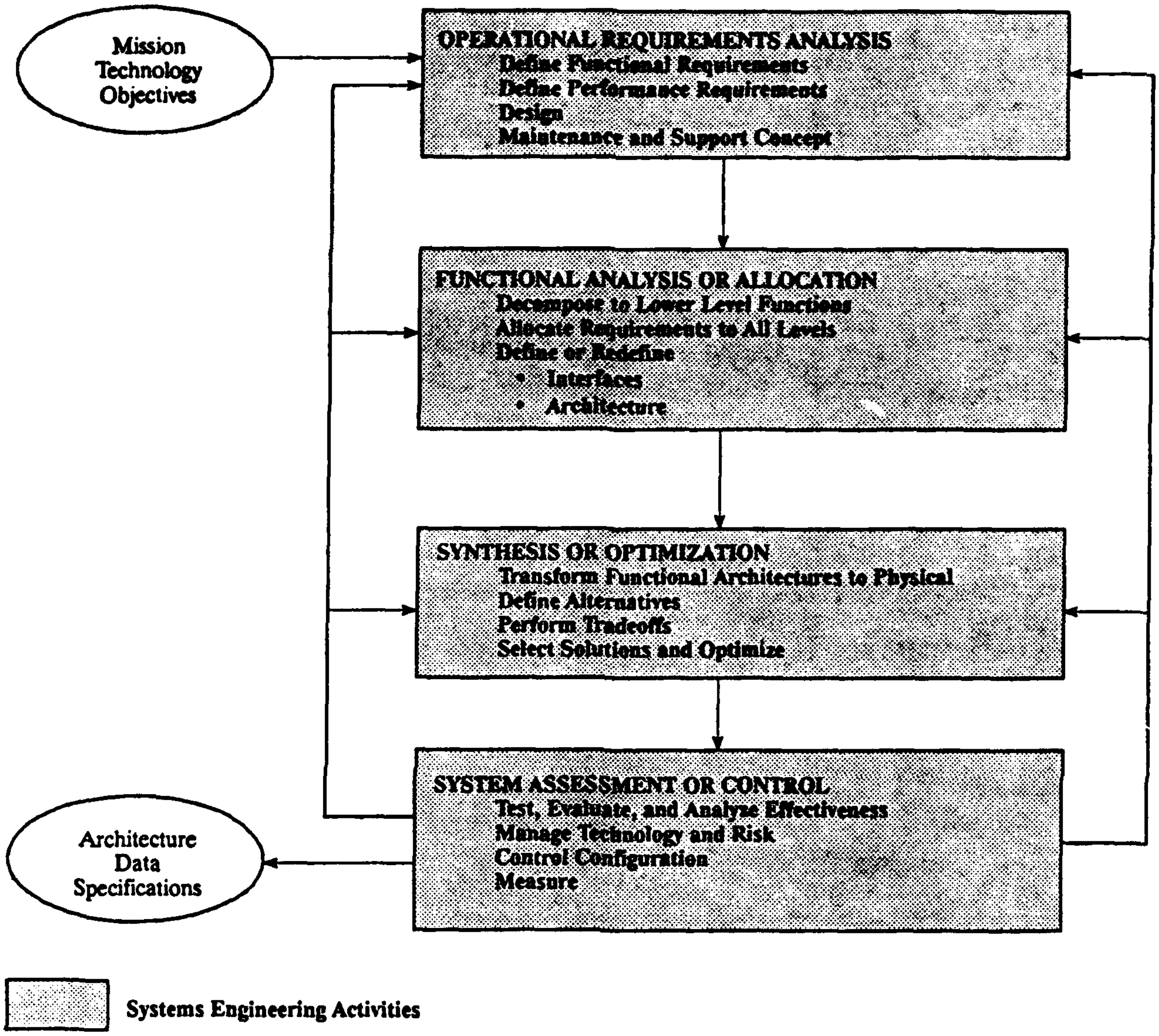

Figure 10. The System Definition Process

LIST OF ABBREVIATIONS AND ACRONYMS

CIM	Corporate Information Management
CM	Configuration management
CMM	Capability Maturity Model
COTS	commercial off-the-shelf software
CSC	computer software component
CSCI	computer software configuration item
CSU	computer software unit
DoD	Department of Defense
GQM	goal-question-metric
HWC	hardware component
HWCI	hardware configuration item
HWU	hardware unit
IEEE	Institute of Electrical and Electronic Engineers
IPT	integrated product team
ISO	International Standards Organization
KSLOC	thousands of source lines of code
LH	labor hours
LM	labor months
NCOSE	National Council on System Engineering
QSM	quantitative system management
RFP	request for proposal

ROI	return on investment
SEI	Software Engineering Institute
SEPG	Software Engineering Process Group
SLOC	source lines of code (also known as source statements)
WBS	work breakdown structure

REFERENCES

Basili, V.R., and D.M. Weiss 1984	A Methodology for Collecting Valid Software Engineering Data. *IEEE Transactions on Software Engineering* SE-10, 6.
Blanchard, B.S. 1991	*System Engineering Management*. New York, New York: John Wiley and Sons.
Blanchard, B.S., and W.J. Fabrycky 1990	*Systems Engineering and Analysis*. 2d ed. Englewood Cliffs, New Jersey: Prentice Hall.
Crosby, P.B. 1980	*Quality Is Free*. New York, New York: Mentor.
Cruickshank, R.D., and M. Lesser 1982	"An Approach to Estimating and Controlling Software Development Costs." In *The Economics of Data Processing*. New York, New York: Wiley.
Department of Defense 1985	*Technical Reviews and Audits for Systems, Requirements, and Computer Programs*, DOD-STD-1521B. Washington, D.C.: Department of Defense.
1988	*Defense System Software Development*, DOD-STD-2167A. Washington, D.C.: Department of Defense.
1991a	*Department of Defense Directive 5000.1*. Washington, D.C.: Department of Defense.
1991b	*Department of Defense Instruction 5000.2*. Washington, D.C.: Department of Defense.
1992a	*Software Development and Documentation*, MIL-STD-SDD(498) (Draft). Washington, D.C.: Department of Defense.
1992b	*Work Breakdown Structure Elements for Software*, MIL-HDBK-171 (Draft). Washington, D.C.: Department of Defense.
1993a	*Military Standard on Work Breakdown Structures for Defense Material Items*. Washington, D.C.: Department of Defense.
1993b	*Military Standard on Systems Engineering*, MIL-STD-499B (Draft). Washington, D.C.: Department of Defense.

IEEE 1992	*Standard for Software Productivity Metrics,* P1045. New York, New York: Institute for Electrical and Electronic Engineers.
Imai, M. 1986	*Kaizen.* New York, New York: McGraw-Hill.
International Organization for Standardization 1990	*Information Technology—Software Product Evaluation—Quality Characteristics and Guidelines for Their Use,* ISO/IEC DIS 9126 (Draft). Geneva, Switzerland: International Organization for Standardization.
Kaplan, R.S. 1988	One Cost System Isn't Enough. *Harvard Business Review* January-February 1988: 61-66.
National Council on Systems Engineering 1993	*Proceedings of the International Symposium.* Washington, D.C.: National Council on Systems Engineering.
Paulk, M.C., Bill Curtis, Mary Beth Chrissis, and Charles V. Weber 1993	*Capability Maturity Model for Software, Version 1.1,* CMU/SEI-93-TR-24. Pittsburgh, Pennsylvania: Software Engineering Institute.
Sage, A.P. 1992	*Systems Engineering.* New York, New York: John Wiley and Sons.
1993	Personal communications with R.D. Cruickshank and R. Werling. August 24, September 8, and September 28, 1993.
Schaffer, R.H. and H.A. Thomson 1992	Successful Change Programs Begin With Results. *Harvard Business Review* January-February 1992.
Software Engineering Institute 1992	*Defining and Using Software Measures.* Pittsburgh, Pennsylvania: Software Engineering Institute.
Strassmann, P.A. 1992	The Policies, Processes, and Technologies of CIM. *Crosstalk* (Software Technology Support Center, Hill Air Force Base, Utah) no. 37 (Oct. 1992): 12-18.
Software Productivity Consortium 1992	*Software Measurement Guidebook,* SPC-91060-CMC, version 02.00.02. Herndon, Virginia: Software Productivity Consortium.
1993a	*System Engineering Workshop Results Summary,* SPC-93128-MC, version 01.00.05. Herndon, Virginia: Software Productivity Consortium.

1993b — *Using New Technologies: A Technology Transfer Guidebook,* SPC-92046-CMC, version 02.00.00. Herndon, Virginia: Software Productivity Consortium.

Weiss, D.M. 1981 — *Evaluating Software Development by Analysis of Change Data,* TR-1120. College Park, Maryland: University of Maryland Computer Science Center.

Werling, R. 1967 — Action-Oriented Information Systems. *Datamation* June 1967.

This page intentionally left blank.