INTRODUCTION

SUMMARY

This Chapter discusses the scope of small space flight projects at Langley Research Center and describes how the systems engineering process is integrated with the Langley project life cycle to achieve project goals in a systematic way.

1.1 PURPOSE

The NPG 7120.5A, NASA Program and Project Management Processes and Requirements, states that a clearly structured and defined systems engineering process will be used in the formulation and implementation of NASA projects to assure that each system achieves its scientific and/or technical goals with demonstrated quality performance and within planned budget and schedules. This handbook describes the systems engineering discipline and tools available to support the advocacy, definition, and development of flight systems at Langley Research Center.

However, the information in this handbook should serve a basic reference for any project to develop a tailored sequence of events, which will lead to achieving the best system design for the project.

This handbook provides a reference for project team members in selecting, tailoring, and implementing a systems engineering process for LaRC projects. This handbook is intended to provide sufficient treatment to allow a comprehensive application of systems engineering for the largest and most complex projects. Depending on the scope of each effort, the process should be tailored to assure the appropriate level of systems engineering application. These determinations should be made at project inception through discussions between the systems engineer and the project or study manager, and, upon approval by Center management, be made a part of the Project Plan. Project personnel may find the handbook useful in defining and planning detailed tasks within the systems engineering context. The handbook also provides a review of related computer tools and glossary terms.

1.2 SCOPE

The procedures described are primarily applicable to small space flight projects that are implemented in-house at LaRC, as defined in LAPD 7120.2, Authority and Responsibilities of Managers of Small Space Flight Projects. In such systems, the primary project components (hardware and software) are developed and integrated in-house, although elements may be contracted to industry. It is intended that this handbook provide the steps necessary for the application of effective systems engineering to these developments.

Similar procedures are used when a system is acquired out-of-house, but this handbook does not address contractor-specific issues. However, even contracted efforts will usually require an in-house systems study to evaluate the progress and results of the contract effort. As a general rule, formulation sub-process studies will be conducted in-house, regardless of the overall acquisition strategy.

Often, a LaRC small space flight project may be part of a larger NASA project managed at another Center. If the other Center requests it, the LaRC project may have to conform to the larger system's requirements.

NASA policies and responsibilities specified in NPG 7120.5A, delegate responsibility for the implementation of a project, including systems engineering and design, to the Project Manager within guidelines and controls described by the funding organization and the LaRC Center Director. This handbook addresses all aspects of systems development, including the creation of hardware and software architectures, and the development and management interfaces between subsystems, from the initiation of the project through flight operations and termination.

New projects are initialized from promising advanced studies or in response to a NASA headquarters request or an Announcement of Opportunity. During the formulation sub-process a study team is formed, at the request of the sponsoring organization, to assess the feasibility of the proposed effort. The systems engineer and the study manager implement the systems engineering process to achieve the goals and present the recommendations of the study team to the sponsoring management.

If approval is given to continue with the sub-process with iterative formulation, a project manager will typically be assigned to direct the expanded effort by the study team. It will be the responsibility of the study team to finalize the recommended Project Plan including a systems engineering management plan defining the systems engineering process to be used. It is intended that the project manager and the systems engineer consider all the systems engineering activities presented in this handbook; however, it is their responsibility to tailor these recommendations as appropriate to their project.

1.3 BACKGROUND

Efforts to implement an effective systems approach to NASA space flight programs are in progress Agencywide. These efforts have resulted in NPG 7120.5A, NASA Program and Project Management Processes and Requirements, and SP-6105, NASA Systems Engineering Handbook. This handbook is consistent with NPG 7120.5A but was tailored for the systems engineering process at LaRC. A selected list of NASA publications is contained in Appendix E.

A system may be defined as a set of interrelated elements organized to work together toward a common goal. Systems may be considered to be the building blocks which comprise projects and programs. Systems can be decomposed into smaller entities that may be considered systems within their defined boundaries. It is this subsystem decomposition, along with careful and complete boundary or interface definition, which allows the partitioning of complex systems into manageable entities. Ultimately, these manageable pieces are represented by those drawings and specifications necessary to construct the operational system, and thus satisfy system technical goals. In addition to these technical aspects, project goals include non-technical goals, such as cost, schedule, and Agency interests. It is the responsibility of the project to develop a system that satisfies the principal customer, while also satisfying ancillary project constraints. Many constraints are external, and out of the control of the project. However, systems engineering has a strong impact on the cost and schedule projections which are the basis for funding, and is accordingly responsible for their accuracy.

A hierarchical system reference is often used for decomposing hardware and software systems down to the lowest level. Such a hierarchy can be used as the basis for the Work Breakdown Structure (WBS) and for subsequent systems decomposition and allocation of requirements. The systems hierarchy shown in Table 1.1 has been proposed as a common NASA terminology for hardware and software system constituents. Each system must be structured with the number and type of levels appropriate for the specific application and consistent with the system external interfaces.

1.4 SYSTEMS ENGINEERING PROCESS

The objective of systems engineering is to provide a robust system that satisfies the customer's technical performance objectives within the constraints of cost and schedule. The challenge is to ensure that a system is developed which meets all imposed requirements and provides the proper balance of system performance, life cycle cost, and development time.

A systems engineering process is the approach to achieve this objective. This process is an orderly sequence of tasks designed to accomplish the optimum design of systems. Optimum does not imply the ultimate in technical performance, the lowest cost, or the shortest schedule. Rather, optimum is defined by the project goals and requirements and will be a balance of these three factors. The systems engineering approach includes the identification of scientific and technical goals; the definition of system constraints and performance measures; the analysis and development of system concepts; and review, verification, and validation of the candidate system(s). During the formulation phases of the project, the systems engineering process relies heavily on an iterative, analytical approach termed the Systems Analysis and Design Procedure. The steps of this procedure are outlined in Table 1.2 and will be described in more detail in Chapter 3. The procedure for applying structured systems engineering to LaRC in-house projects is discussed in Chapter 4 and outlined in Chapter 6.

Table 1.1 - System Hardware and Software Hierarchies

SYSTEMS HIERARCHY

Program
Project
System
Segment

Software	Hardware
Computer Software Configuration Item (CSCI)	Element
	Subsystem
Computer Software Component (CSC)	Assembly
Computer Software Unit (CSU)	Subassembly
	Part

Table 1.2 - Steps in the Systems Analysis and Design Procedure.
1. Initialization: The initial step is to organize the study and acquire the required tools and resources support. In a LaRC development, this step assures that new projects have been reviewed by the Sponsoring Group Director and that appropriate resources and skills are available to achieve the goals of the current project phase. Typically, the start of a new project phase requiring NASA Headquarters funding will be closely related to the NASA budget cycle. Important systems engineering factors to be addressed

	include an assessment of the objectives for the phase, technical approaches, and analytical tools to be used.
2.	User Needs and Goals Analysis: The major product of this step will be the Goals Analysis Document and Hierarchy, which will initially be in a preliminary form and will gradually be refined into subsequent phases. The goals document is the basis for subsequent system metrics such as performance measures, requirements flow down, and verification and validation standards. The systems engineering function must monitor system performance through all project phases to assure that system goals will be achieved.
3.	Systems Requirements and Constraints: This step establishes the constraints at each phase leading to progressively more detailed definition of requirements for the hardware and software architectures. During the early phases, system requirements are developed which satisfy the system goals. As the system design progresses, hardware and software requirements are allocated to the segments of the system and progressively lower in the hierarchy. An important systems engineering function is to track the allocated requirements through final verification and maintain traceability to the system requirements.
4.	Performance Measures: Performance measures are parameters that are defined to provide criteria for subsequent systems analyses and trade studies. For product developments, these are the variables used to judge the overall attractiveness of system candidates and a subset of the project requirements. Initially, performance measures are the parameters used for system selection and then they evolve, along with constraints, into requirements or systems validation criteria.
5.	Systems Concepts: Alternative solutions are generated in this step. Initially, the output of this activity is a set of candidate system options for analysis and trade-off studies. In Phase B, the list of alternate concepts is reduced and the baseline system concept is selected. Requirements and resources are allocated and "design to" specifications are defined for the segments. In project implementation, these segment requirements become the standard for the application of systems engineering analysis at the segment level. This decomposition process is repeated throughout the systems hierarchy until concepts are defined down to a level which allows for easy piece part selection and computer software unit development.
6.	Concepts Analysis: This step analyzes and defines the performance of alternate systems concepts so that comparisons can be made for final system concept selection. Mathematical modeling and computer simulation are frequently employed during this step. Analytical tools are used throughout the project life cycle to estimate and verify systems performance.
7.	Concepts Ranking: During this step the set of systems concepts are ranked by decision analysis in order of their overall performance (including cost and

schedule) so that selection of preferred concepts can be made. The resulting Alternate Concepts Analysis Document becomes the basis for the configuration management plan that is utilized throughout the project life cycle.
8. Systems Development: This step incorporates all of the detailed development activities required to advance the systems concept(s) and design to the desired level of maturity for the current phase of the project. These activities include design refinements and risk reduction activities that were identified during concept analysis.
9. Review, Verification, and Validation: The Systems Analysis and Design Procedure is iterative and requires ongoing, critical examination of efforts. During this step, the systems concepts are subjected to verification processes to assure system integrity and to external reviews to secure management concurrence and commitment for the project to continue. The verification process is employed throughout the project life cycle.
10. Decision Point: This important step or control gate will provide management direction for the future of the project; approval to proceed to the next phase, direction to go back and revise the project approach, or a decision to terminate the project effort.

1.5 PROJECT LIFE CYCLE

A new project is born out of recognition of a need or opportunity which addresses NASA's goals and missions. If the project appears to be promising, it is selected by the LaRC sponsoring group for further definition and a Pre-Phase A system study is undertaken to evaluate feasibility. A typical project passes through defined phases or cycles as it proceeds from conceptual definition through design, fabrication, integration, and test to an operational role. Progression from one phase to the next is dependent upon meeting the phase objectives and passing prescribed reviews and decision points. The total sequence is termed the "project life cycle." The progressive phases in the NASA project life cycle are listed in Table 1.3. The life cycle may be tailored to meet the unique requirements of a specific project.

Table 1.3 - NASA Project Life Cycle

1. Pre-Phase A: ADVANCED STUDIES - Preliminary Requirements and Concepts Analysis
2. Phase A: PRELIMINARY ANALYSIS - Requirements Definition and Conceptual Trade Studies
3. Phase B: DEFINITION - Concept Definition and Preliminary Design (Source selection, if required)
4. Phase C: DESIGN - Final Design and Engineering Development
5. Phase D: DEVELOPMENT - Fabrication, Integration, Test, and Evaluation
6. Phase E: OPERATIONS - Preflight and Flight Mission Operations and Disposal

1.6 SYSTEMS ANALYSIS AND DESIGN PROCEDURE MODEL

The Systems Analysis and Design Procedure addresses the detailed activities and products which must be accomplished to support project decisions concerning the system under development. Systems engineering analysis is inherently an iterative process resulting in successive refinement of the system design through each phase of the project. The Systems Analysis and Design Procedure is heavily utilized by the project team in the early formulation phases (Pre-Phase A, Phase A, and Phase B) of the project, and is used in support of subsystem level trades as the development progresses. In the implementation phases (Phase C, Phase D, and Phase E) of the project, systems engineering is more heavily involved with integration and verification activities and the Systems Analysis and Design Procedure is less formalized. In the closing activities of the implementation phases, systems engineering attempts to validate system performance against customer requirements and record lessons learned from the project. It is important to note the clear distinction between systems engineering, which is concerned with all aspects of system development, and the Systems Analysis and Design Procedure, which is a tool utilized by engineers for optimization.

The steps of the Systems Analysis and Design Procedure are combined with the project life cycle in the schematic model shown in Figure 1.1. The process starts at the center of the circle when Pre-Phase A activities are initiated and follows a clockwise path through each sector of the circle as each task of the Systems Analysis and Design Procedure is addressed.

When the circuit is completed the option exists to repeat the cycle (or specific tasks) for better definition or to pass through the decision point to the next phase of the project. Each formulative phase of the project is addressed using the same systematic approach. As the project moves from phase to phase, the tasks also evolve to address the changing objectives as will be shown in Chapter 6.

This Systems Analysis and Design Procedure model has several distinctive features that can help in visualizing the overall project development process:

- The process is initiated by a formal management decision to select a promising candidate concept or idea for further study and investigation. There may be many research proposals under consideration at any given time, but only a few can be developed into potential projects. When a candidate project is advocated by management for in-depth study, this establishes priority, level of effort of personnel, and associated schedule for a systems study.

- The model illustrates a specific sequence of steps to be followed during the systems study. The process steps are completed consecutively in a logical progression during each phase of the study, as appropriate. This provides a focused and structured method that will result in the most efficient approach to the study. It is expected that the Systems Analysis and Design Procedure will be guided by the systems engineer, who will direct and coordinate the effort to assure that the study goals are achieved.

- The process steps outline a problem-solving approach rather than a fixed step-by-step procedure. Each step may be tailored to accomplish the defined objectives of the current phase of the project. For example, during the early formulative phases of the project (Pre-Phase A, Phase A, and Phase B), the emphasis is on system level analysis and design tasks. In the later implementation phases (Phase C and Phase D), the focus is on detailed design of segments and lower levels of assembly. The steps may be repeated iteratively to achieve the design maturity desired during the phase. In the event of a change in project goals, the process may be repeated to define the impact of changes.

- Each phase of the process is a distinct entity as indicated by the concentric, labeled circles. There may be circumstances in a project where certain areas lead others; for example, long lead items which require longer to develop. In general, the project should remain focused on the objectives of each particular phase until that phase is completed. As noted on the model, transition from one phase to another requires passage through a decision point or control gate. This assures management concurrence and support. Each phase is a distinct activity and successful completion of a phase demonstrates that the project is showing progress toward accomplishing its end goals.

The general flow of the systems engineering process for a hypothetical LaRC project is displayed in Figure 1.2. This figure shows how the systems engineering process and the Systems Analysis and Design Procedure may be applied within the context of project life cycle phases and control gates. The generic steps of the procedure including goals analysis, systems requirements, concepts analysis and ranking, development, and so forth, are shown within each phase. The iterative nature of the process is also emphasized by the provision for repeating the cycle for improved resolution of the products. The process is repeated in subsequent phases as applicable with appropriate changes in emphasis to continue refinement of the system. This evolutionary approach will be developed in more detail in Chapter 4.

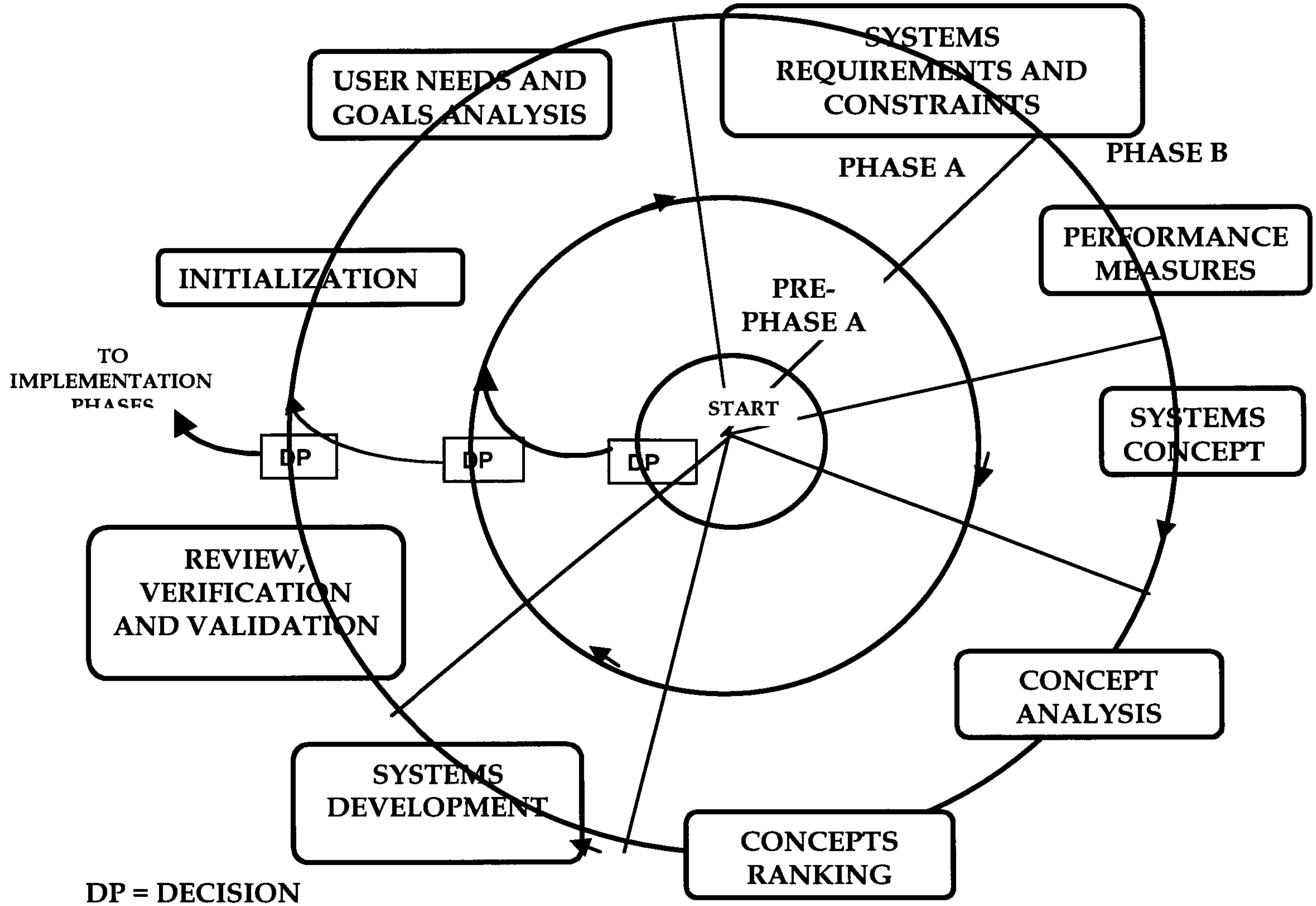

Figure 1-1 – Systems Analysis and Design Procedure Model.

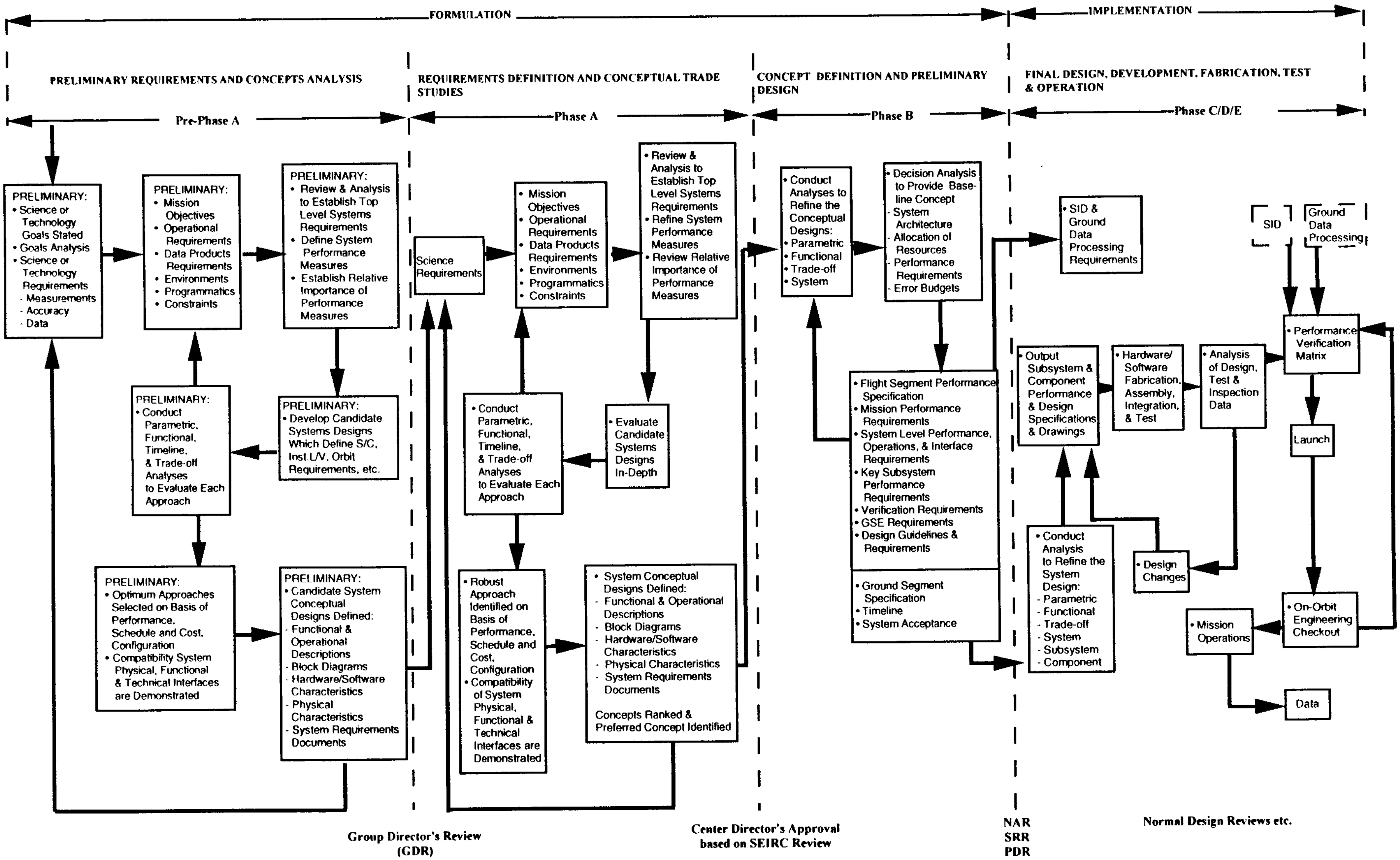

Figure 1.2 - Flow of the Systems Engineering Process

Chapter 2

SYSTEMS ENGINEERING MANAGEMENT

SUMMARY

This Chapter gives an overview of the interaction between project management and systems engineering in the Langley Research Center project environment.

2.1 PROJECT MANAGEMENT

Within the NASA context, the Project Manager is defined by NPD 7120.4A, "Program/Project Management," as "the senior official at the NASA field installation exclusively responsible for managing execution of the project life cycle to accomplish program objectives within guidelines and controls prescribed by program and field installation management." The Langley Research Center (LaRC) Project Manager reports through Center management to the Program Manager at NASA Headquarters or at the host field installation which has overall program responsibility. Thus, the Project Manager has the key responsibility to be the LaRC focal point for the project.

A major responsibility of project management is to control the project resources within the three constraints of time, cost, and performance. This task is ultimately the responsibility of the Project Manager. The Project Manager also is responsible for developing the overall Project Plan and for the day-to-day responsibilities of managing the project team to achieve the goals that have been defined.

One role of the Project Manager (or Pre-Phase A Study Manager) is to establish a project environment which is supportive of the systems engineering task. This role includes setting project priorities and ensuring needed resources are available. The Project Manager is also responsible for resolving problems identified by the systems engineer and project team. At the beginning of each phase of the project, the Project Manager, with the systems engineer, must prioritize the systems engineering tasks and define the project objectives for that phase.

2.2 SYSTEMS ENGINEERING RESPONSIBILITIES

Within the project organization, the Project Manager typically delegates systems engineering and technical responsibility to the systems engineer. In this position, the systems engineer has technical direction over all the engineering disciplines and must coordinate all project activities within the context of the systems engineering process. Just as the Project Manager's focus is primarily on overall project management and controlling the project resources, the systems engineer's emphasis and responsibility is to assure that the system accomplishes its technical purpose in the most cost-effective way. The Project Manager's role and the systems engineer's role are mutually dependent and the effectiveness of the project effort hinges on their interaction.

At LaRC, the systems engineer's role will be assigned to different individuals, depending upon the project organization. The systems engineer may have a title such as Instrument Manager (IM), Technical Project Engineer (TPE), Instrument Project Engineer (IPE), or Systems Engineer. The term "systems engineer," as used in this handbook, describes the individual responsible for managing the systems engineering process.

A study or project will usually have only one systems engineer assigned during all project phases. However, larger projects which require a broad range of technical expertise may need systems engineers assigned to different segments of the system. All of these individuals should be familiar with the systems engineering process as implemented on the project.

The systems engineer has overall technical accountability for the design and operation of all systems on the project. Detailed project activities are actually performed by the cognizant project discipline, that is, design engineering, test engineering, software engineering, fabrication, assembly, and so forth. In terms of the three project restraints of time, cost, and performance, systems engineering is primarily responsible for system performance. System performance also impacts time and cost; therefore, the systems engineer must continually address all three project restraints.

The systems engineer is responsible to the Project Manager and is usually the focal point of the project for all technical performance at the systems level. The Project Manager specifically addresses project management interfaces and specialty functions such as cost, schedule control, safety, program assurance, and configuration management. The systems engineer is concerned with all the aspects of systems engineering management including: baseline management, requirements review and traceability, system specifications, change control, design reviews, audits, document control, failure review boards, control gates, and performance certification. The systems engineer is also concerned with the internal and external system technical interfaces. It is the systems engineer's responsibility to assure that all of the systems perform properly when the system is fully integrated.

The systems engineer must have broad experience in many technical disciplines and should have relevant systems experience with flight hardware and software in the project environment. The systems engineer must also be able to work effectively with the project team to accomplish the systems engineering tasks for each project phase. The composition of the project team by phase will be discussed in detail in Chapter 4. The

the project team by phase will be discussed in detail in Chapter 4. The systems engineer may also serve as the Contracting Officer's Technical Representative (COTR) for contracted efforts.

2.3 SYSTEMS ENGINEERING MANAGEMENT

This handbook is to be used as a guide for each project to provide a planned application of systems engineering which is appropriate to the effort. The systems engineering management plan is the project document that defines how the project systems engineering function will be technically managed within the constraints established by the Project Plan. For a large project, a formal, documented systems engineering management plan may be used to describe how the systems engineering effort will be managed. The document may be organized in three parts:

Part I - <u>Technical Program Planning and Control</u>

The first part defines organizational responsibilities and addresses such issues as configuration, documentation, design control methods, and review requirements.

Part II - <u>Systems Engineering Process</u>

The second part describes the process to be used, risk management approach, types of mathematical simulation models, and other application related information.

Part III - <u>Engineering Specialty Integration</u>

The third part describes the integration of the specialty engineering disciplines, project approach to concurrent engineering, verification and validation, and so forth.

Chapter 3

SYSTEMS ANALYSIS AND DESIGN PROCEDURE

SUMMARY

This Chapter describes the overall philosophy of the Systems Analysis and Design Procedure with details on application during the critical, early Formulation Phases and in the later Implementation Phases of the project.

3.1 INTRODUCTION

This Chapter describes the overall philosophy of the Systems Analysis and Design Procedure; details on implementation in each phase will be provided in Chapter 6. The process is a simple set of steps, applicable to any problem, and is designed to find the best solution, versus an adequate solution. "Best," in this context, is from the point of view of technical performance, as well as cost and schedule. The process begins deductively, from the general to the specific, by establishing the broad top goal of the project and working toward the details. As the project progresses, "bottom-up" work is accomplished to compliment the initial deductive nature. However, the fundamental concept requires that the overall picture is established first, before detailed actions (for example, hardware piece part or software module design) are undertaken. The goals established initially in the process serve as a clear direction for project work and establish the criteria for success in the form of constraints and variables for optimization, typically referred to as requirements. After establishing these performance requirements, options are explored and ranked numerically. All facets of the process are repeated, or iterated, for error control, leading to successive refinement of systems goals and requirements. The following sections describe the Systems Analysis and Design Procedure in detail.

3.2 INITIALIZATION

This precursor step is necessary to assure that appropriate resources and skills are available to achieve the immediate phase goals within the specified schedule and cost constraints. It is necessary that the systems engineering function be properly planned prior to initiation of the study. Relevant factors include an assessment of the goals for the study, organizational responsibilities, technical approach, and analytical tools to be used. It is the responsibility of the systems engineer to assure that the study goals will be achieved on a timely basis. Any expected deviations from these goals should be immediately brought to the attention of the sponsoring Group Director.

Products of the initialization step are typically work and staffing plans and detailed schedules to meet the near term milestones. Formal decision points (control gates) may be supplemented by more detailed entrance criteria to be satisfied before work can commence at the start of the cycle. Exit criteria, in the form of required activities and products, may also be established to assure that key elements will be completed prior to the end of the Systems Analysis and Design Procedure cycle for the phase.

3.3 USER NEEDS AND GOALS ANALYSIS

This endeavor is perhaps the most crucial undertaking of the project, since it is here that success is defined. However, it is insufficient to simply define success, because the definition is unclear or ambiguous. The steps of the goals analysis are designed to provide a coherent, complete set of objectives for the project team. These steps are:

3.3.1 Top Level Goal

The first concern of the goals analysis is to obtain a single, top level goal for the undertaking. This should be a short, concise statement of what the project hopes to accomplish and should exclude, to the greatest extent possible, any details as to how it will be achieved. "How" qualifiers are, in effect, constraints that limit the possible alternatives. Typically, the top level goal is available at the onset; but it is still necessary to assure the accuracy and validity of the project goal statement. This is best accomplished by putting this goal in context with other broader goals of the organization such as vision statements, policy, or organizational thrusts. Care should be taken to assure that this goal is consistent with other endeavors and to clearly emphasize that the project is addressing a new area of concern. Discussions should be conducted with the customer/user of the output to define the central issue of the project and to assure that project goals are consistent with defined mission needs. Typical considerations may be:

*Why is the goal as stated?
*Are there underlying goals not brought to light?
*Is the statement the true goal or rather a means of achieving another goal?

The purpose of this activity is to assure that the true goal is defined. The total project team should be a part of these discussions to assure a thorough understanding, but ultimately, the definition of the goal statement is the responsibility of the customer/user or, typically, the Principal Investigator. The systems engineer typically serves as a facilitator or coordinator in establishing the top level goal. Participation of a systems engineering facilitator acting as a third party in these discussions is beneficial since queries may be made in a non-threatening manner by a third party with no stake in the outcome.

3.3.2 Current Status

Once the top level goal statement is established, research is conducted to determine and record the current state of the art or status of efforts addressing the goal. Other current work which relates to the project, both internal and external to the organization, is recorded. A history of the events leading to the current situation is also helpful in preventing a repeat of past mistakes. Specific attention should be paid to the problems or shortcomings currently associated with the endeavor, along with the efforts under way to resolve those problems. Usually, the customer is the best source of this information, which may easily be obtained through discussions. However, further independent attempts to gather data are well advised as they may reveal information from unexpected sources. This may be done through library searches or by contacting identified external sources directly to query their knowledge of the situation. Conversations with those associated with similar efforts provide an excellent means for establishing a complete picture of the current state of the art. Emphasis should be placed on positive and negative aspects of the status quo. Any and all pertinent details are compiled in order to prevent duplication of effort and give the team a good understanding of the starting point of the project. The systems engineer should summarize this information in the Statement of Project Status. Given a firm foundation of the current standing, the next step follows directly.

3.3.3 Vision

The project vision is a set of statements that describes the desired situation upon successful completion of the project. This narrative describes the positive aspects of the current status, plus improvements made and a report of those problems or shortcomings which have been overcome. These discussions may be used to stimulate ideas that are technical advances beyond the current situation or outside the current paradigms associated with the project. Beyond this, the strength of the vision is to provide the project team with a clear and common direction to their efforts. To be effective, the vision should be established with the customer, reviewed with team experts, and circulated among the team members in a high profile manner. The vision must be something that the team can become excited about and support, but at the same time, be realistic and feasible within the constraints and available resources of the organization.

3.3.4 External Factors

Time is taken during the goals analysis for explicit discussions of the external political factors affecting the project and the likely implications to the development. Typically, these discussions will include such topics as NASA Headquarters sponsorship, funding trends, related administrative policy, impact on the public, and existing or likely competitors. The purpose is to explicitly address the unwritten rules which will affect the project and to develop plans for clarifying, qualifying, and reducing any related risk from external constraints. Uncertain political situations should be a part of the overall project risk assessment as they may have a greater impact than the technical risks. Consideration should be given to other groups affected by fulfillment of the top goal for the purpose of developing a constituency or advocacy group. The political advantages of pursuing the project goal as stated, as well as the drawbacks, should be considered. The attitudes and opinions of those with decision-making authority over the project must be carefully considered. If opposition appears unmanageable, a modification of the goals may be in order.

3.3.5 Goals Tree

The culmination of the goals analysis is the development of the hierarchy of project goals, or the goals tree. This is a graphic depiction of the goals structure, with appearance similar to an organization chart, which shows the top-level goal and its relationship to more specific supporting goals. The top-level goal, as previously developed, is displayed at the top of the tree, with supporting goals below. Those in a supporting level are referred to as sub-goals which "will support" the accomplishment of the goal at the next higher level. Each sub-goal is further refined to a subsequent level which "will support" its fulfillment. A branch is terminated when the sub-goal meets the qualifications for requirements (see Section 3.4) which assures that the sub-goal may be indisputably judged as met (or failed). As an illustration, a sub-goal which was worded "to provide for minimum delay in data transfer" would be insufficiently defined to warrant termination of that branch of the tree. This sub-goal would require further breakdown and may terminate the branch with a statement such as "to transfer 230 kilobytes of information to the ground station within 15 seconds." It should be noted that even at the terminating nodes of the tree, the requirements should describe precisely what is to be accomplished without describing how the accomplishments are to be made. "To be determined" (TBD) is a useful placeholder for numerical entries, but should be replaced with target values as soon as possible.

One branch from a goals tree is shown in Figure 3.1, "Sample Branch from Typical Goals Tree." The progressive decomposition from the top level goal through sub-goals to the specific science functional requirements at the lowest level is clearly illustrated. The shaded boxes are the "leaf" terminations of the branch.

The significance of this process is that system requirements are tied directly to the top-level goal of the project.

Goals Hierarchy for LIDAR at LaRC

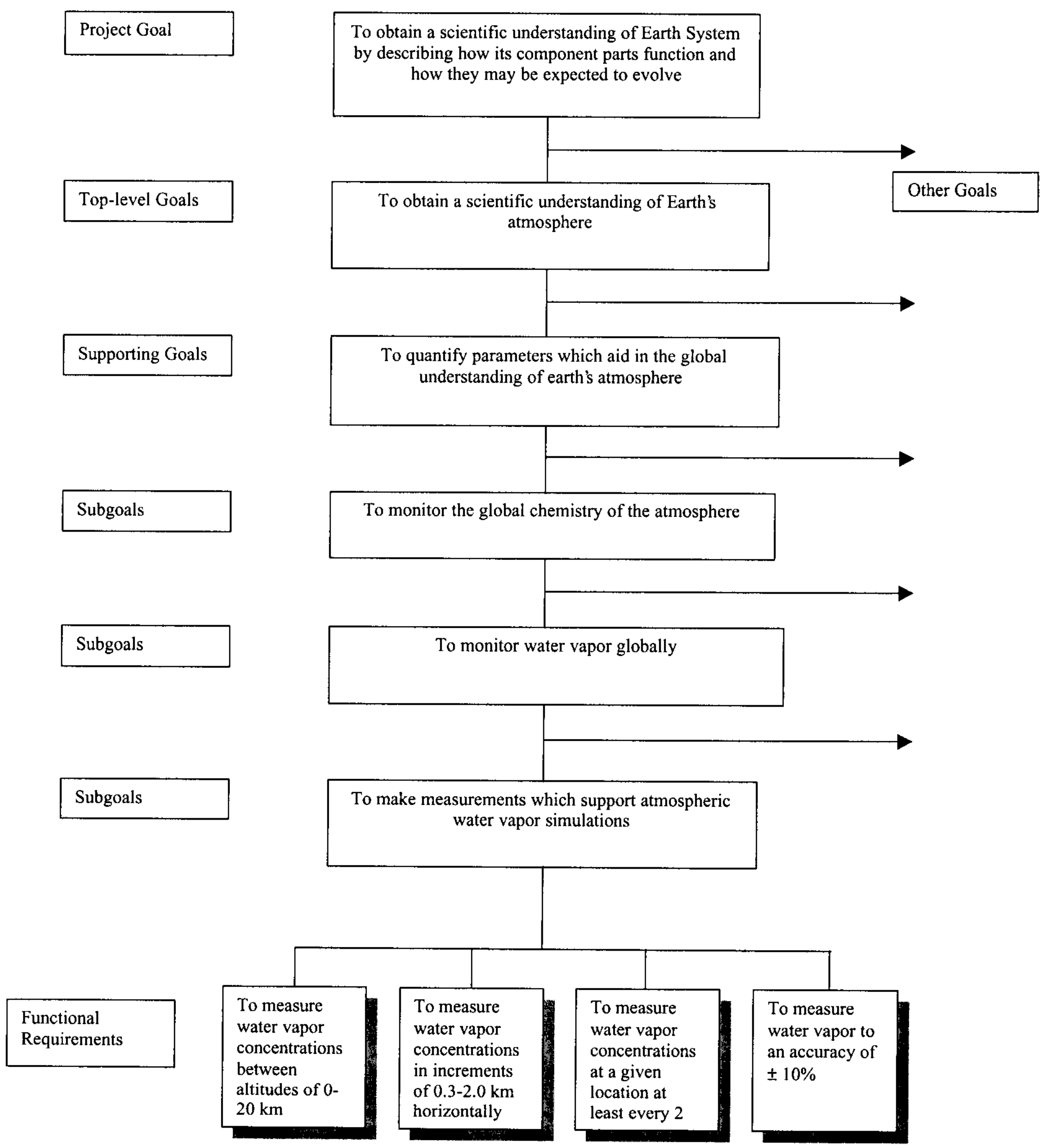

Figure 3.1 – Sample Branch from Typical Goals Tree.

3.3.6 Validation of Goals

To this point, the goals analysis has been an outscoping exercise that has solicited input without criticism. Next, the work is reviewed and scrutinized for error, completeness, and consistency. Where criticism was previously discouraged, it is now invited. Compatibility with the Langley Research Center (LaRC) vision and the extent to which the project fulfills a NASA Headquarters strategic plan milestone should be assessed. After this review, the goals analysis process is iterated so that any points which surfaced during the subsequent application may be fed back into the appropriate steps. The first iteration should be a rough cut to get the pertinent issues out into the open. This may typically be accomplished in a day of meetings. The information must then be organized by the systems engineer for presentation to the team at the next meeting.

3.3.7 Summary

The product of this step will be the Goals Analysis Document which will initially be issued in a preliminary form and gradually refined as the process is repeated in the later phases of the project life cycle. The goals document will form the basis for all subsequent project activities including requirements flow down, verification, and validation standards.

3.4 SYSTEMS REQUIREMENTS AND CONSTRAINTS

This step of the Systems Analysis and Design Procedure defines requirements and constraints at each phase of the life cycle leading to progressively more detailed definition of functional requirements for the hardware and software architectural hierarchies. The input for this activity is the terminating branches of the goals tree. Essentially, these functional requirements, or objectives, define what must be done to achieve project goals. They provide a quantitative description of measurable entities. Requirements may be considered as one of two types: constraints or performance measures. Constraints limit the set of possible options by establishing system boundaries. Performance measures are the variables used for system and subsystem trade-off analysis. An example of a constraint would be spacecraft launch weight which must be kept below a certain mass for a particular orbital condition. Performance measures are discussed in Section 3.5. All types of requirements must meet certain necessary qualifications. By definition, requirements must include, as a minimum, the properties described in Sections 3.4.1 through 3.4.5.

3.4.1 Quantifiability

Some numeric value must be assignable to the requirement. The quantifiable nature of a requirement is the attribute which allows the determination of success. Requirements such as "minimum downtime" or "maximum data rate" cannot be verified without further definition. The requirements must be put into the form of a numeric value such as ". . .

downtime of less than 100 seconds per 30 days" or ". . . data rate of greater than or equal to 5 kilobits per second." Just as in cost and schedule requirements, there must be some quantitative measure associated with each parameter.

3.4.2 Objectivity vs. Subjectivity

This means that the definition of terms is the same for everyone. An example of a quantifiable, but subjective measure, may be "the number of acceptable data points must be greater than . . ." The term "acceptable" is not an objective standard.

3.4.3 Meaningfulness

This qualifier is added to assure that measures directly address the goals of the project, as opposed to choosing criteria simply because they may be easily counted or for personal reasons. Deductive requirement development through the goals tree is the best method of excluding spurious requirements.

3.4.4 Absolute Scale

Requirements should not be measured relative to some other parameter that may vary as this leads to confusion in the assessment of the satisfaction of requirements. An example of a relativistic requirement may be ". . . shall be degraded by less than 20 percent due to solar activity."

3.4.5 Verifiability

This is an extremely important requirement characteristic since any requirement that cannot be verified by some method should be discarded as meaningless. Note that typical methods of verification include analysis, as well as inspection and test.

3.4.6 Summary

The product of this effort is the Systems Requirements Document and, ultimately, the requirements database for the project. The Systems Requirements Document will include all of the systems requirements including the instrument, mission, spacecraft, and operations requirements. The requirements definition will become progressively more refined during each project phase and these systems requirements will eventually be flowed down through the hardware and software hierarchy to the appropriate levels.

3.5 PERFORMANCE MEASURES

Performance measures are defined during the goals analysis to provide criteria for subsequent trade studies of alternate system configurations. A subset of the project objectives, performance measures, are variable parameters used to provide a quantitative description of the system's ability to satisfy functional goals. The parameters will usually have a limit on the minimum acceptable level of performance, but achievement beyond that limit is of significant value. For example, while an instrument may have a minimum performance accuracy of 90 percent, an option providing 95 percent accuracy might be more desirable and worthy of additional cost. This is in contrast to constraint type objectives such as system volume, in which no real advantage is realized by performing better than the required value. The performance measures are typically ranked to provide an indication of their relative importance. The results of this process are compiled in the Performance Measures Statement.

3.6 SYSTEMS CONCEPTS

Ideally, any and all potential system concepts should be considered in order to describe the entire range of possible solutions. However, to prevent being overcome by a large number of possibilities, classes of concepts have sufficient definition for initial studies. As analyses continue, clear front runners emerge in each of the feasible classes, which are studied in greater detail.

The most widely used technique for identifying options is brainstorming, but more all inclusive techniques, such as combinatorics, may be used to generate alternatives. A combining approach first breaks the general concept into subsystems, elements, or functional segments, and then considers the possibilities on each decomposed piece. The various options are then recombined to form specific system concepts. This may often bring to light concepts that were not previously considered. The output of this activity is a set of candidates that are described with only the essential details. A quick review will likely remove some options as unfeasible, or obviously incapable of satisfying constraints. The surviving alternate concepts then progress to the next step in the process, in which candidate performance is predicted. As the project progresses, the list of alternate concepts is reduced until the baseline system concept is selected in Phase B. As the system is decomposed into smaller pieces, the process is repeated for alternative element, subsystem, assembly, and subassembly evaluation and selection.

3.7 CONCEPTS ANALYSIS

Once the set of possible concepts has been developed, performance of each must be projected as an input to decisions. The previously established performance measures define precisely what performance must be projected. The concept characterizations are the "bottom-up" or inductive complement to the deductive work accomplished in the goals analysis. Estimates are sought for each feasible solution (one which meets constraints) on the value for each performance measure. A closer look will likely identify additional concepts which cannot meet the minimum requirements.

Mathematical modeling and computer simulation are very useful at this stage, especially for new developments about which there is no existing performance information. Probability distributions may be even more useful than point estimates of variables since they also describe the uncertainty associated with each projection. Known values are recorded at their specific values. Note that cost and schedule projections of the alternative concepts are always required as a minimum. This analysis of concepts and performance measure projection allows for a structured approach to determining the overall best solution, through use of decision analysis.

The product of this effort is the Alternate Concepts Analysis Document which provides a comparison of each of the alternate systems. The alternate concepts are eventually reduced to the baseline system concept during Phase B. More detailed analysis of concepts is typically done as the system is decomposed down to the subsystem, assembly, and lower levels.

3.8 CONCEPTS RANKING

This step is the culmination of the procedure in which the set of systems concepts is ranked through decision analysis for the purpose of indicating their overall attractiveness. The inputs to decision analysis are the performance measures and their relative weights, the set of feasible alternative concepts options, and the estimations for each option on each performance measure. The output is a list of surviving solutions in numerically ranked order.

To accomplish the ranking, the performance measure estimates that each concept is first normalized to a value between 0 and 1. The normalized numbers are combined with performance measure weights to produce an overall score for each concept. The advantage of a structured decision approach is that a very difficult, multiple attribute decision may be broken down into a number of more simple, single factor judgments. In a more sophisticated implementation, it is possible to express the variation in a decision maker's utility over the range of possible values of a performance measure. Also, a number of models exist for the overall combination of the performance measure weights and concept values. Regardless of the approach taken, the project will be structured in a fashion to allow for easy analysis of the solutions.

Since there is subjectivity in the weights of performance measures, a variation in these values will reveal the impact on the order of preference of the candidate concepts. A robust solution, one which is relatively insensitive to variations in the performance measure weights, is preferred. Sensitivity analysis may be used to graphically display the effects of variation in performance measure weights. Example decision analyses are shown in Appendix D.

3.9 SYSTEMS DEVELOPMENT

In the formulation phases, concepts are designed and breadboard or brassboard hardware and prototype software are developed in support of risk reduction activities. During the implementation phases, the conceptual layouts of Pre-Phase A, Phase A, and Phase B are formally designed, fabricated (or procured), and integrated and tested. The sequence of development may include prototype or engineering model hardware with supporting software. In all phases the development effort may be subject to changes in schedule, changes in funding profile, or technical performance required. In addition, during the verification process, it may be discovered that the technical performance desired is unachievable under time and money constraints. Renegotiations among the project, funding bodies, and customer will be necessary in these cases to decide on an altered approach; for example, seek additional funding, allow schedule slip, or accept lesser technical performance. Frequent team meetings and status reports from subsystem managers are crucial to identify problems before they are unachievable within project constraints. Likewise, extreme attention must be paid to the definition and maintenance of interfaces among subsystems to allow for easy integration. The systems engineer must be continually alert to the occurrence of increased risk in the system. The sequence of development is usually driven by external constraints or risk mitigation.

3.9.1 Technical Risk Management

Beginning in Pre-Phase A of the project, risk aspects of the development must be identified, characterized, and mitigated to an acceptable level. The systems engineer is responsible for reduction of technological risk and for development of the Risk Reduction Plan. For LaRC, this risk is inherent in most projects because of the research orientation and lack of previously demonstrated performance. Thus, a systematic approach must be undertaken to productively reduce the project technological uncertainties. Risk (or expected outcome) is defined as the severity of the occurrence of an event, multiplied by the probability of that event or:

$$\text{Risk(Event A)} = p(A) * \text{Cost}(A)$$

where $p(A)$ is a value between 0 and 1 and $\text{Cost}(A)$ is usually described in dollars. This returns the value of Risk(Event A) in expected cost to the project. Thus, it may be seen that catastrophic hazards may (and usually are) acceptable if the probability of the event is sufficiently low.

Conversely, a relatively innocuous event may require risk reduction if the probability of the event is high. The approach to risk reduction consists of three steps: identification, quantification, and reduction as outlined below.

3.9.1.1 Risk Identification

The best methods of risk identification include brainstorming (with experts in the field of endeavor) and review of lessons learned. As in other inductive techniques, this exercise should begin without criticism in order to explore all possible sources of risk. Subsequent quantification will quickly identify those risks worth tracking. A list of technical risk areas should be compiled and maintained for the life of the project. As risk areas are reduced in value, they may be removed from the list of those currently under review, but should be maintained for the project audit trail, and to serve as lessons learned for future projects.

3.9.1.2 Risk Quantification

This step is necessary in order to determine which of the identified risks require action. Expected costs may be determined as above, or more sophisticated calculation techniques may be utilized if the data is obtainable. A slightly better approach is to estimate the most pessimistic, most optimistic, and most likely costs, should the hazard occur. These values may be weighted and added as follows:

Risk(Event A) = [2/3 * Cost(A likely)]+
[1/6 * Cost(A optimistic)]
[1/6 * Cost(A pessimistic)]

This approach is sufficient for most applications.

If probability distributions are available (for example exponential, Weibull, and so forth) then the functions may be combined as:

Risk(Event A) = †0* xf(x)dx

where:

x = Cost of Event A
f(x) = Probability of Event A

Like the two other methods, this equation will return the expected cost of Risk (Event A).

A qualitative approach to risk assessment may also be taken. In this case, the judgment of experts is used to prioritize risk areas based on past experience. This identifies a rank order of risk events needing attention, which may be subjected to risk reduction and tracked to closure.

3.9.1.3 Risk Reduction

After the risk events are quantified, they must be examined to assess their acceptability. For those risks considered too severe, structured steps must be taken to reduce or remove the impact of the event. It is important to note that in practical cases, there should not be more efforts expended on risk reduction than the expected value of the risk itself. However, if the risk could lead to total system failure, the cost to reduce the risk may be justifiable up to the total cost of the system, given the probabilities are high. In these cases, additional research is usually indicated to mature the concept before the project development begins.

The two basic strategies for reduction of risk are to decrease the probability or decrease the severity of the risky event. The following paragraphs illustrate approaches to risk reduction from both of these approaches:

*Decrease the Probability

If the possibility exists to delete that portion of the system responsible for a given risk (in favor of a less risky alternative), this should be considered first. Obviously, in many cases this will not be feasible. For example, the limited life span of detectors cannot be removed from system risk by eliminating the use of detectors, since they are fundamental to system operation. Rather, the approach may be to develop or qualify certain long-life detectors to satisfy system requirements. Another approach may be to strengthen the integrity of the weak points of a design by utilizing higher reliability components. In the event that alterations to the system are not feasible, some additional control may be possible through the introduction of procedures that are implemented to heighten confidence in the integrity of the system.

*Decrease the Severity

An alternative to the above is modification of the system design to include backup systems. This should lead to a so-called "graceful degradation" in which a component failure will lead to a decrease in overall performance, but will not result in a total system failure. An example may be seen in satellite communications, where a failure of the primary system would result in operation of a backup transponder. The backup may move much less data, but would still be sufficient to transmit the most critical information. Another possibility may be to introduce warning systems that alert the system operator of impending problems. This approach is valid only if the mechanism and time exists to alleviate the problem before a catastrophic failure occurs.

3.9.2 Other Risk Categories

The risk related to items within the boundary of the systems under development is addressed above. However, there are other types of risk that are imposed on the project by external concerns. As a minimum, the systems engineer must be concerned with safety risk and schedule risk. Safety risk is concerned with control of events that pose a hazard to entities outside the system boundary; most notably personnel and ancillary equipment. For LaRC projects, the reduction of safety risk will be overseen by a product assurance engineer who is responsible for the imposition of safety oriented requirements. Typically, safety risk is mitigated through regulations or constraints imposed by the organization responsible for the launch of the system. For many LaRC flight projects, these requirements are levied by the National Space Transportation System (NSTS) safety organizations at Johnson Space Center (JSC) and Kennedy Space Center (KSC). In a similar fashion, safety requirements are established by the parties responsible for alternate launch vehicles and launch facilities.

Schedule risk must be dealt with directly by the systems engineer. Schedule risk may be associated with development time for an item or technology on the critical path, with procurement time for a critical item, or with personnel scheduling problems. This risk is described in expected system delivery slip in units of time. High schedule risk events are typically addressed through contingency planning and the development of technologies which are parallel to the baseline approach.

3.10 REVIEW, VERIFICATION, AND VALIDATION

As outlined in Section 3.4, requirements must have some associated method of verification. Typically, the term verification relates to system specific hardware and software requirements. In contrast, validation is concerned with assuring that the system meets the needs of the customer. Thus, system validation truly occurs only in flight. However, while complete system testing prior to flight will not prove the capability to meet customer needs, integrated system verification can certainly identify the inability to do so.

In the system life cycle, verification usually begins in risk reduction efforts. Certain portions of the conceptual system are built or modeled in an effort to verify their ability to satisfy certain hardware or software requirements. Life testing may be indicated in the early project stages to establish expected life of an unproven design. Beyond risk reduction, verification occurs as the first portions of the engineering model are received or assembled. This usually assesses the ability of a subsystem to meet its performance as required in its anticipated environment. This testing will continue at higher levels as larger portions of the system are assembled. Ultimately, this process leads to system level verification and validation in flight.

Review is a constant element of the systems engineering process for the purpose of error control and improvement. The steps of the process are performed as applicable in each project phase and iterated to assure the accuracy of the products. This internal iteration occurs on a daily basis within the project. The formal, external reviews occur as control gates between phases.

3.11 DECISION POINT

The culmination of the Systems Analysis and Design Procedure for each phase is the decision made on the basis of the phase study products. In an LaRC development, this decision will occur at the end of each phase of project formulation when a review is convened to assess the readiness of the project to proceed. This review process allows the customer to assess the state of the project and make decisions concerning future directions. While the result of convening a control gate review may be to proceed without condition or to cancel the effort, the outcome will typically fall somewhere in between. Additional work, as required by control gate action items, will usually be required before proceeding to the next phase. Regardless of the final outcome, control gates should be treated as true decision points relative to the future of the project, and not simply an exercise for the project team.

Continue to Next Section