https://ntrs.nasa.gov/search.jsp?R=20170007239 2019-08-31T06:54:12+00:00Z

NASA/SP-2016-6105-SUPPL

Expanded Guidance for NASA Systems Engineering

Volume 2: Crosscutting Topics, Special Topics, and Appendices

National Aeronautics and Space Administration

NASA Headquarters

Washington, D.C. 20546

March, 2016

Part 2 Table of Contents

Part 2 Table of Figures

Part 2 Table of Tables

Part 2 Table of Blue Boxes

7.0 Crosscutting Topics

The topics in this chapter cut across all life-cycle phases and are of special interest for enhancing the performance of the systems engineering process or constitute special considerations in the performance of systems engineering. These topics include the following:

- Engineering with contracts: applying systems engineering principles to contracting and contractors;
- Concurrent engineering methods: diverse specialists systematically collaborating simultaneously in a shared environment, real or virtual, to yield an integrated design;
- Selecting engineering design tools: integrated design facilities and tools;
- Environmental, nuclear safety, and planetary protection policy compliance: protecting the environment and discussing the importance of the Nation's space assets;
- Use of the metric system;
- Systems engineering on multi-level/multi-phase programs and projects: special considerations;
- Fault management: understanding and managing the off-nominal system behaviors;
- Technical margins: establishing and managing for contingencies to reduce development risk and increase the chance for mission success; and
- Human systems integration: balancing total system safety and effectiveness to ensure mission success.

7.1 Engineering with Contracts

7.1.1 Introduction, Purpose, and Scope

Historically, most successful NASA projects have depended on effectively blending project management, systems engineering, and technical expertise among NASA, contractors, and third parties. Underlying these successes are a variety of agreements (e.g., contract, memorandum of understanding, grant, cooperative agreement) between NASA organizations or between NASA and other Government agencies, Government organizations, companies, universities, research laboratories, and so on. To simplify the discussions, the term "contract" is used to encompass these agreements.

This section focuses on the NASA systems engineering activities pertinent to awarding a contract, managing contract performance, and completing a contract. In particular, NASA systems engineering interfaces to the procurement process are covered, since the NASA engineering technical team plays a key role in the development and evaluation of contract documentation.

Contractors and third parties perform activities that supplement (or substitute for) the NASA project technical team accomplishment of the NASA common systems engineering technical process activities and requirements outlined in this guide. Since contractors might be involved in any part of the systems engineering life cycle, the NASA project technical team needs to know how to prepare for, allocate or perform, and implement surveillance of technical activities that are allocated to contractors.

7.1.2 Acquisition Strategy

While this section pertains to projects where the decision has already been made to have a contractor implement a portion of the project, it is important to remember that the choice between "making" a product in-house by NASA and "buying" it from a contractor is one of the most crucial decisions in systems development. (See Section 5.1.) Questions that should be considered in the "make/buy" decision include the following:

- Is the desired system a development item or more off-the-shelf?
- What is the relevant experience of NASA versus potential contractors?
- What are the relative importance of risk, cost, schedule, and performance?
- Is there a desire to maintain an "in-house" capability?
- What portion(s) of the total system life cycle will the contracted work address and how will this contracted portion be made to mesh with NASA "in-house" activities or with portions performed on separate contracts? How will the various pieces be made to work together efficiently and cost-effectively and who has responsibility for integration throughout the program/project life cycle?

As soon as it is clear that a contract will be needed to obtain a system or service, the responsible project manager should contact the local procurement office. The contracting officer will assign a contract specialist to navigate the numerous regulatory requirements that affect NASA

procurements and guide the development of contract documentation needed to award a contract. The contract specialist engages the local legal office as needed.

7.1.2.1 Develop an Acquisition Strategy

The project manager, assisted by the assigned procurement and legal offices, first develops a project acquisition strategy or verifies the one provided. The acquisition strategy provides a business and technical management outline for planning, directing, and managing a project and obtaining products and services by contract.

In some cases, it may be appropriate to probe outside sources in order to gather sufficient information to formulate an acquisition strategy. This can be done by issuing a Request for Information (RFI) to industry and other parties that may have interest in potential future contracts. An RFI is a way to obtain information about technology maturity, technical challenges, capabilities, price and delivery considerations, and other market information that can influence strategy decisions.

The acquisition strategy includes the following:

- Objectives of the acquisition—capabilities to be provided, major milestones;
- Acquisition approach—single acquisition or a series of acquisitions, single or multiple suppliers/contracts, competition or sole source, funding source(s), phases, system integration, Commercial Off-The-Shelf (COTS) products;
- Business considerations—constraints (e.g., funding, schedule), availability of assets and technologies, applicability of commercial items versus internal technical product development;
- Risk management of acquired products or services—major risks and risk sharing with the supplier;
- Contract types—performance-based or level of effort, fixed-price or cost reimbursable;
- Contract elements—incentives, performance parameters, rationale for decisions on contract type; and
- Product support strategy—oversight of delivered system, maintenance, and improvements; i.e., if the contract is for only portions of the product life cycle—e.g., product development alone—how will operations be managed?

The technical team gathers data to facilitate the decision-making process regarding the above items. The technical team knows about issues with the acquisition approach, determining availability of assets and technologies, applicability of commercial items, issues with system integration, and details of product support. Similarly, the technical team provides corporate knowledge to identify and evaluate risks of acquiring the desired product, especially regarding the proposed contract type and particular contract elements.

7.1.2.2 Acquisition Life Cycle

Contract activities are part of the broader acquisition life cycle, which comprises the phases of solicitation, source selection, contract monitoring, and acceptance. (See Figure 7.1-1.) The

acquisition life cycle overlaps and interfaces with the systems engineering processes in the project life cycle. Acquisition planning focuses on technical planning when a particular contract (or purchase) is required. (See Section 6.1.) In the figure below, requirements development corresponds to the Technical Requirements Definition Process in the systems engineering engine. (See Figure 2.1-1.) The next four phases—solicitation, source selection, contract monitoring, and acceptance—are the phases of the contract activities. If the contract is for acquisition of a major product (e.g., a space vehicle), transition to operations and maintenance represents activities performed to transition the acquired product(s) to the organization(s) responsible for operating and maintaining them (which could be separate and/or follow-on contractor(s)). If achieving the results of the overall program (e.g., "human crew access to low Earth orbit") will involve a mix of in-house and contracted responsibilities, clear authority, roles and responsibilities should be assigned to the integrating agents. The term "acquisition management" is often used to refer to crosscutting program/project management activities that are performed throughout the system acquisition life cycle as accepted by the acquiring organization or allocated to other entities.

Figure 7.1-1 Acquisition Life Cycle

7.1.2.3 NASA Responsibility for Systems Engineering

The NASA technical team is responsible for systems engineering throughout the acquisition life cycle. The technical team contributes heavily to systems engineering decisions and results, whatever the acquisition strategy, for any combination of suppliers, contractors, and subcontractors. The technical team is responsible for systems engineering regardless of whether the acquisition strategy calls for the technical team, a prime contractor, or some combination of the two.

This subsection provides specific guidance on how to assign responsibility for surveillance measures when translating the technical processes onto a contract. Generally, the Technical Planning, Interface Management, Technical Risk Management, Configuration Management, Technical Data Management, Technical Assessment, and Decision Analysis Processes should be implemented throughout the project by both the NASA team and the contractor. Stakeholder Expectations Definition, Technical Requirements Definition, Logical Decomposition, Design Solution Definition, Product Implementation and Integration, Product Verification and Validation, Product Transition, and Requirements Management Processes are implemented by NASA or the contractor depending upon the level of the product decomposition. When written and before execution, contracts should be clear on roles and responsibilities, especially on the specifics of NASA's tight engagement with the contractor during design and development. Without clear allocation of surveillance interactions and deliverables during contract negotiations, NASA might forfeit having a role in quality assurance of the end product(s).

Table 7.1-1 provides guidance on how to implement the 17 technical processes from NPR 7123.1. The first two columns have the number of the technical process and the requirement statement of responsibility. The next column provides general guidance on how to distinguish

who has responsibility for implementing the process. The last column provides a specific example of the application of how to implement the process for a particular project. The particular scenario is a science mission where a contractor is building the spacecraft, NASA assigns Government-Furnished Property (GFP) instruments to the contractor, and NASA operates the mission.

7.1.3 Prior to Contract Award

7.1.3.1 Acquisition Planning

Based on the acquisition strategy, the NASA technical team needs to plan acquisitions and document the plan in developing the project's SEMP. The SEMP covers the NASA technical team's roles, responsibilities, and involvement in the periods before contract award, during contract performance, and upon contract completion. Included in acquisition planning are solicitation preparation, source selection activities, contract phase-in, monitoring contractor performance, acceptance of deliverables, completing the contract, transition beyond the contract, and overall program integration. The SEMP focuses on interface activities with the contractor, including NASA technical team involvement with and monitoring of contracted work.

Often overlooked in project staffing estimates is the amount of time that NASA technical team members are involved in contracting-related activities. Depending on the type of procurement, a technical team member involved in source selection could be consumed nearly full time for 6 to 12 months. After contract award, NASA technical monitoring consumes 30 to 50 percent, peaking at fulltime when critical milestones or key deliverables arrive. Keep in mind that for most contractor activities, NASA staff performs supplementary activities. To ensure efficient and effective use of the NASA team during contract monitoring, acceptance of deliverables, contract completion and transition, and overall program integration, it is critical that sufficient and appropriate metrics, milestones, and reporting requirements are levied on the contractor. Time and attention of the NASA technical team to the contract's contents prior to contract execution are critical to ensuring that the contractor will provide NASA with sufficient information during the contract's execution for NASA to fulfill oversight responsibility. During contract execution, contractor and NASA technical teams should work together to update the SEMP at appropriate milestones.

Solicitations

The release of a solicitation to interested parties is the formal indication of a future contract. A solicitation conveys sufficient details of a Government need (along with terms, conditions, and instructions) to allow prospective contractors (or offerors) to respond with a proposal. Depending on the magnitude and complexity of the work, a draft solicitation may be issued. After proposals are received, a source evaluation board (or committee) evaluates technical and business proposals per its source evaluation plan and recommends a contractor selection to the contracting officer. The source evaluation board, led by a technical expert, includes other technical experts and a contracting specialist. The source selection process is completed when the contracting officer signs the contract.

The most common NASA solicitation types are RFP and Announcement of Opportunity (AO). Visit the online NASA Procurement Library for a full range of details regarding procurements and source

Table 7.1-1 Applying the Technical Processes on Contract

#	NPR 7123.1 Process	General Guidance on Who Implements the Process	Application to a Science Mission as Example
1	[3.2.1 Stakeholder Expectations Definition Process] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for the definition of stakeholder expectations for the applicable WBS model.	There will generally be a set of stakeholders at the NASA level for the end product. For lower-level products, stakeholders could be at either NASA or the contractor. If stakeholders are at the contractor, then the contractor should have responsibility and vice versa.	Stakeholders for the mission/project are within NASA; stakeholders for the spacecraft power subsystem are mostly at the contractor.
2	[3.2.3 Technical Requirements Definition Process] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for definition of the technical requirements from the set of agreed-upon stakeholder expectations for the applicable WBS model.	The SEMP should identify the level at which the responsibility for developing requirements would transition to a contractor. Assignment of responsibility generally follows the stakeholders, e.g., if stakeholders are at the contractor, then requirements are developed by the contractor and vice versa.	NASA develops the high-level requirements, and the contractor develops the requirements for the power subsystem.
3	[3.2.4 Logical Decomposition Process] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for logical decomposition of the validated technical requirements of the applicable WBS.	Follows the requirements, e.g., if requirements are developed at the contractor, then the decomposition of those requirements is implemented by the contractor and vice versa. In the case at the contractual boundary, the requirements would be the responsibility of NASA and decomposition by the contractor.	NASA performs the decomposition of the high-level requirements down to the contractual boundary. The contractor performs the further decomposition of the power subsystem requirements.
4	[3.2.5 Design Solution Definition Process] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for designing product solution definitions within the applicable WBS model that satisfy the derived technical requirements.	Follows the requirements, e.g., if requirements are developed at the contractor, then the design of the product solution is implemented by the contractor and vice versa. In the case at the contractual boundary, the requirements would be the responsibility of NASA and design by the contractor.	NASA designs the mission/project, and the contractor designs the power subsystem.

#	NPR 7123.1 Process	General Guidance on Who Implements the Process	Application to a Science Mission as Example
5	[3.2.6 Product Implementation Process] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for implementation of a design solution definition by making, buying, or reusing an end product of the applicable WBS model.	In general, follows the design, e.g., if the design is developed at the contractor, then the implementation of the design is performed by the contractor and vice versa. However, there are cases where NASA may generate a design and have it implemented by a contractor. Responsibilities should be defined in the SEMP.	NASA implements (and retains responsibility for) the design for the mission/project, and the contractor does the same for the power subsystem.
6	[3.2.7 Product Integration Process] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for the integration of lower level products into an end product of the applicable WBS model in accordance with its design solution definition.	Follows the design, e.g., if the design is developed at the contractor, then the integration of the design elements is performed by the contractor and vice versa. Sometimes NASA serves as the final integrator for the end product.	NASA integrates the design for the mission/project, and the contractor does the same for the power subsystem.
7	[3.2.8 Product Verification Process] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for verification of end products generated by the Product Implementation Process or Product Integration Process against their design solution definitions.	Follows the product integration, e.g., if the product integration is implemented at the contractor, then the verification of the product is performed by the contractor and vice versa.	NASA verifies the mission/project, and the contractor does the same for the power subsystem.
8	[3.2.9 Product Validation Process] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for validation of end products generated by the Product Implementation Process or Product Integration Process against their stakeholder expectations.	Follows the product integration, e.g., if the product integration is implemented by the contractor, then the validation of the product is performed by the contractor and vice versa. For the case of the contractual boundary, NASA may choose to perform the validation.	NASA validates the mission/project, and the contractor does the same for the power subsystem.
9	[3.2.10 Product Transition] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for transitioning end products to the next-higher-level WBS model customer or user.	Follows the product verification and validation, e.g., if the product verification and validation is implemented by the contractor, then the transition of the product is performed by the contractor and vice versa.	NASA transitions the mission/project to operations, and the contractor transitions the power subsystem to the spacecraft level.

#	NPR 7123.1 Process	General Guidance on Who Implements the Process	Application to a Science Mission as Example
10	[3.2.11 Technical Planning Process] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for planning the technical effort.	Assuming both NASA and the contractor have technical work to perform, then both NASA and the contractor need to plan their respective technical efforts.	NASA would plan the technical effort associated with the GFP instruments and the launch and operations of the spacecraft, and the contractor would plan the technical effort associated with the design, build, verification and validation, and delivery and operations of the power subsystem.
11	[3.2.12 Requirements Management Process] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for management of requirements defined and baselined during the application of the system design processes.	Follows process #2. Responsibility for approving requirement changes should be identified in the SEMP.	
12	[3.2.13 Interface Management Process] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for management of the interfaces defined and generated during the application of the system design processes.	Interfaces should be managed one level above the elements being interfaced.	The interface from the spacecraft to the project ground system would be managed by NASA, while the power subsystem to attitude control subsystem interface would be managed by the contractor.
13	[3.2.14 Technical Risk Management Process] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for management of the technical risk identified during the technical effort.	Technical risk management is a process that needs to be implemented by both NASA and the contractor. All elements of the project need to identify their risks and participate in the project risk management process. Deciding which risks to mitigate, when, at what cost is generally a function of NASA project management.	NASA project management should create a project approach to risk management that includes participation from the contractor. Risks identified throughout the project down to the power subsystem level and below should be identified and reported to NASA for possible mitigation.
14	[3.2.15 Configuration Management] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for CM.	Like risk management, CM is a process that should be implemented throughout the project by both the NASA and contractor teams.	NASA project management should create a project approach to CM that includes participation from the contractor. The contractor's internal CM process will have to be integrated with the NASA approach. CM needs to be implemented throughout the project down to the power subsystem level and below.

#	NPR 7123.1 Process	General Guidance on Who Implements the Process	Application to a Science Mission as Example
15	[3.2.16 Technical Data Management Process] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for management of the technical data generated and used in the technical effort.	Like risk management and CM, technical data management is a process that should be implemented throughout the project by both the NASA and contractor teams.	NASA project management should create a project approach to technical data management that includes participation from the contractor. The contractor's internal technical data process will have to be integrated with the NASA approach. Management of technical data needs to be implemented throughout the project down to the power subsystem level and below.
16	[3.2.17 Technical Assessment Process] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for making assessments of the progress of planned technical effort and progress toward requirements satisfaction.	Assessing progress is a process that should be implemented throughout the project by both the NASA and contractor teams.	NASA project management should create a project approach to assessing progress that includes participation from the contractor. Typically this would be the project review plan. The contractor's internal review process will have to be integrated with the NASA approach. Technical reviews need to be implemented throughout the project down to the power subsystem level and below.
17	[3.2.18 Decision Analysis Process] The Center Directors or designees establish and maintain a process to include activities, requirements, guidelines, and documentation for making technical decisions.	Clearly technical decisions are made throughout the project both by NASA and contractor personnel. Certain types of decisions or decisions on certain topics may best be made by either NASA or the contractor depending upon the Center's processes and the type of project.	For this example, decisions affecting high-level requirements or mission success would be made by NASA and those at the lower level, e.g., the power subsystem that did not affect mission success would be made by the contractor.

The technical team is intimately involved in developing technical documentation for the acquisition package. The acquisition package consists of the solicitation (e.g., Request for Proposals (RFPs)) and supporting documents. The solicitation contains all the documentation that is advertised to prospective contractors (or offerors). The key technical sections of the solicitation are the SOW (or performance work statement), technical specifications, and contract data requirements list. Other sections of the solicitation include proposal instructions and evaluation criteria. Documents that support the solicitation include a procurement schedule, source evaluation plan, Government cost estimate, and purchase request. Input from the technical team will be needed for some of the supporting documents.

It is the responsibility of the contract specialist, with input from the technical team, to ensure that the appropriate clauses are included in the solicitation. The contract specialist is familiar with requirements in the Federal Acquisition Regulation (FAR) and the NASA FAR Supplement (NFS) that will be included in the solicitation as clauses in full text form or as clauses incorporated by reference. Many of these clauses relate to public laws, contract administration, and financial management. Newer clauses address information technology security, data rights, intellectual property, new technology reporting, and similar items. The contract specialist stays abreast of updates to the FAR and NFS. As the SOW and other parts of the solicitation mature, it is important for the contract specialist and technical team to work closely to avoid duplication of similar requirements.

7.1.3.2 Develop the Statement of Work

Effective surveillance of a contractor begins with the development of the SOW. The technical team establishes the SOW requirements for the product to be developed. The SOW contains process, performance, and management requirements the contractor should fulfill during product development (see Section 6.1.2, Statement of Work).

As depicted in Figure 7.1-2, developing the SOW requires the technical team to analyze the work, performance, and data needs to be accomplished by the contractor. The process is iterative and supports the development of other documentation needed for the contracting effort. The principal steps in the figure are discussed further in Table 7.1-2.

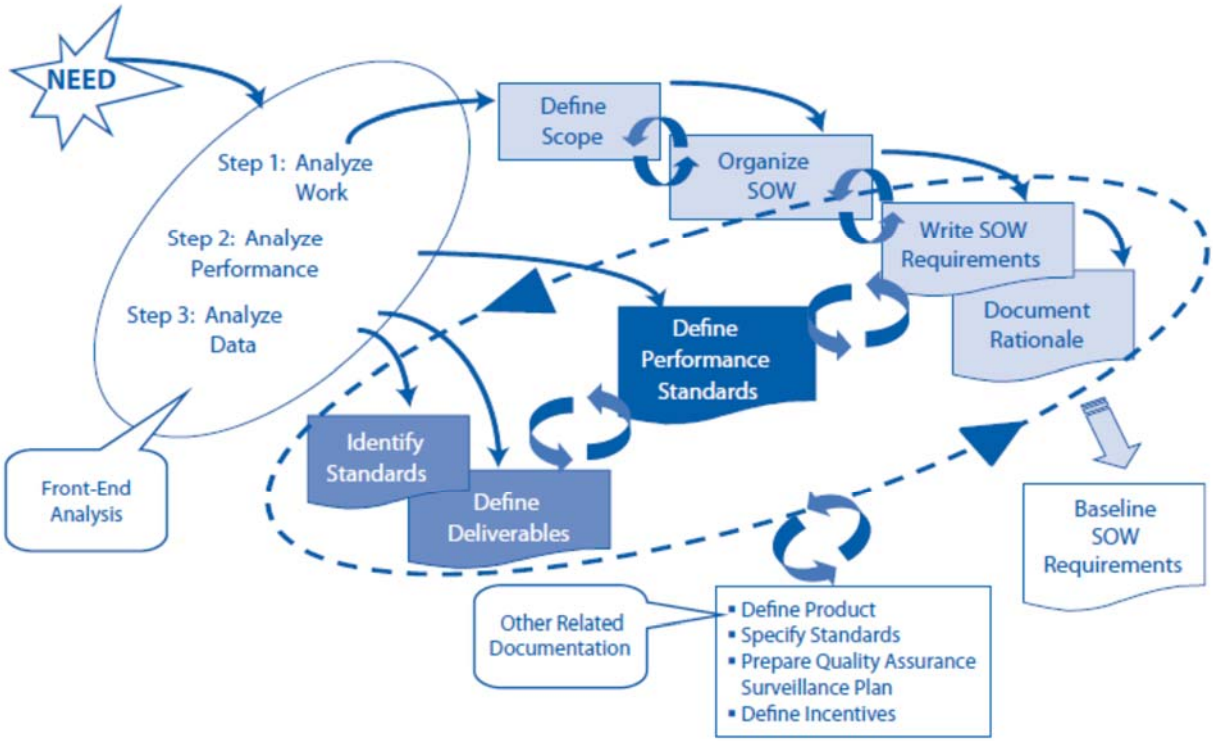


Figure 7.1-2 Contract Requirements Development Process

Table 7.1-2 Steps in the Requirements Development Process

Step	Task	Detail
Step 1: Analyze the Work	Define scope	Document in the SOW that part of the project's scope that will be contracted. Give sufficient background information to orient offerors.
	Organize SOW	Organize the work by products and associated activities (i.e., product WBS).
	Write SOW requirements	Include activities necessary to: Develop products defined in the requirements specification; and Support, manage, and oversee development of the products. Write SOW requirements in the form "the contractor shall." Write product requirements in the form "the system shall."
	Document rationale	Document separately from the SOW the reason(s) for including requirements that may be unique, unusual, controversial, political, etc. The rationale is not part of the solicitation.
Step 2: Analyze Performance	Define performance standards	Define what constitutes acceptable performance by the contractor. Common metrics for use in performance standards include cost and schedule. For guidance on metrics to assess the contractor's performance and to assess adherence to product requirements on delivered products, refer to *System and Software Metrics for Performance-Based Contracting*.
Step 3: Analyze Data	Identify standards	Identify standards (e.g., EIA, IEEE, ISO) that apply to deliverable work products including plans, reports, specifications, drawings, etc. Consensus standards and codes (e.g., National Electrical Code, National Fire Protection Association, American Society of Mechanical Engineers) that apply to product development and workmanship are included in specifications.
	Define deliverables	Ensure each deliverable data item (e.g., technical data—requirements specifications, design documents; management data—plans, metrics reports) has a corresponding SOW requirement for its preparation. Ensure each product has a corresponding SOW requirement for its delivery.

After a few iterations, baseline the SOW requirements and place them under configuration management. (See Section 6.5.)

Use the SOW checklist in appendix P to help ensure that the SOW is complete, consistent, correct, unambiguous, and verifiable. Below are some key items to require in the SOW:

- Technical and management deliverables having the highest risk potential (e.g., the contractor SEMP, HSI Plan, development and transition plans); requirements and architecture specifications; test plans, procedures and reports; metrics reports; delivery, installation, and operations and maintenance documentation.
- Contractual or scheduling incentives in a contract should not be tied to the technical milestone reviews. These milestone reviews (for example, SRR, PDR, CDR, etc.) enable a critical and valuable technical assessment to be performed. These reviews have specific entrance criteria that should not be waived. The reviews should be conducted when these criteria are met, rather than being driven by a particular schedule.

- Timely electronic access to data, work products, and interim deliverables to assess contractor progress on final deliverables.
- Provision(s) to flow down requirements to subcontractors and other team members.
- Content and format requirements of deliverables in the contract data requirements list. These requirements are specified in a data requirements document or data item description, usually as an attachment. Remember that you need to be able to edit data deliverables.
- Metrics to gain visibility into technical progress for each discipline (e.g., hardware, software, thermal, optics, electrical, mechanical). For guidance on metrics to assess the contractor's performance and to assess adherence to product requirements on delivered products, refer to *System and Software Metrics for Performance-Based Contracting*.
- Metrics that document and track life-cycle reliance on personnel for total system performance. The intent is to ensure that design and development decisions that result in or depend on human involvement (operations, logistics, maintenance, etc.) are being tracked, bound to goals set early in the program, and don't result in life-cycle cost growth discovered late in the program.
- Quality incentives (defect, error count, etc.) to reduce risk of poor quality deliverables. Be careful because incentives can affect contractor behavior. For example, if you reward early detection and correction of software defects, the contractor may expend effort correcting minor defects and saving major defects for later.
- Expectation that use of COTS products is subject to NASA review. (See Section 7.1.3.6.)
- A continuous management program to include a periodically updated risk list, joint risk reviews, and vendor risk approach.
- Surveillance activities (e.g., status meetings, reviews, audits, site visits) to monitor progress and production, especially access to subcontractors and other team members.
- Specialty and crosscutting engineering capabilities (e.g., reliability, quality assurance, cryogenics, pyrotechnics, biomedical, waste management) that are needed to fulfill standards and verification requirements.
- Provisions to assign responsibilities between NASA and contractor according to verification, validation, or similar plans that are not available prior to award.
- Provisions to cause a contractor to disclose changing a critical process. If a process is critical to human safety, require the contractor to obtain approval from the contracting officer before a different process is implemented.

Note: If you neglect to require something in the SOW, it can be costly to add it later.

The contractors should supply a SEMP that specifies their systems engineering approach for requirements development, technical solution definition, design realization, product evaluation, product transition, human systems integration, and their technical planning, control, assessment, and decision analysis. It is best to request a preliminary contractor SEMP in the solicitation. The source evaluation board can use the contractor's SEMP to evaluate the offeror's understanding of the requirements, as well as the offeror's capability and capacity to deliver the system. After contract award, the technical team can eliminate any gaps between the project's SEMP and the

contractor's SEMP that could affect smooth execution of the integrated set of common technical processes.

Often a technical team has experience developing technical requirements, but little or no experience developing SOW requirements. If you give the contractor a complex set of technical requirements but neglect to include sufficient performance measures and reporting requirements, you will have difficulty monitoring progress and determining product and process quality. Understanding performance measures and reporting requirements will enable you to ask for the appropriate data or reports that you intend to use.

Traditionally, NASA contracts require contractors to satisfy requirements in NASA policy directives, NASA procedural requirements, NASA standards, and similar documents. These documents are almost never written in language that can be used directly in a contract. Too often, these documents contain requirements that do not apply to contracts. It is important to understand what the requirements mean and if they apply to contracts. The requirements that apply to contracts need to be written in a way that is suitable for contracts. Alternatively, the SOW might allow the contractor to propose its own versions of certain NASA procedural requirements that meet NASA's intent. The SOW should explain that such contractor-proposed requirements are subject to review and approval by NASA technical and program management teams.

7.1.3.3 Task Order Contracts

Sometimes, the technical team can obtain engineering products and services through an existing task order contract. The technical team develops a task order SOW and interacts with the contracting officer's technical representative to issue a task order. Preparing the task order SOW is simplified because the contract already establishes baseline requirements for execution. First-time users need to understand the scope of the contract and the degree to which delivery and reporting requirements, performance metrics, incentives, and so forth are already covered. Task contracts offer quick access (days or weeks instead of months) to engineering services for studies, analyses, design, development, and testing and to support services for configuration management, quality assurance, maintenance, and operations. Once a task order is issued, the technical team performs engineering activities associated with managing contract performance and completing a contract (discussed later) as they apply to the task order.

7.1.3.4 Quality Assurance Surveillance Plan

The Quality Assurance Surveillance Plan (QASP) defines the monitoring of the contractor effort and is developed at the same time as the SOW. It is critical to the success of the surveillance plan that all expectations, roles, and responsibilities—NASA and contractor—are clearly defined up front. As noted earlier, if a requirement is left out, it can be extremely costly to add it later. (See Figure 2.5-3.) The NASA technical team works with mission assurance personnel (generally from the local Safety and Mission Assurance (SMA) organization) and with systems engineers to prepare the surveillance plan for the contracted effort. Whether performed by NASA technical experts or contractors, mission assurance should be engaged from the start. Prior to contract award, the surveillance plan is written at a general level to cover the Government's approach to perceived programmatic risk. After contract award, the surveillance plan describes in detail

inspection, testing, and other quality-related surveillance activities that will be performed to ensure the integrity of contract deliverables, given the current perspective on programmatic risks.

Recommended activities to include in the surveillance plan follow:

- Review key deliverables within the first 30 days to ensure adequate startup of activities.
- Conduct contractor/subcontractor site visits to monitor production or assess progress.
- Evaluate effectiveness of the contractor's systems engineering processes.

Drafting the surveillance plan when the SOW is developed promotes the inclusion of key requirements in the SOW that enable activities in the surveillance plan. For example, in order for the technical team to conduct site visits to monitor production of a subcontractor, then the SOW should include a requirement that permits site visits, combined with a requirement for the contractor to flow down requirements that directly affect subcontractors.

7.1.3.5 Writing Proposal Instructions and Evaluation Criteria

Once the technical team has written the SOW, the Government cost estimate, and the preliminary surveillance plan, the solicitation can be developed. Authors of the solicitation should understand the information that will be needed to evaluate the proposals and write instructions to obtain specifically needed information. In a typical source selection, the source selection board evaluates the offerors' understanding of the requirements, management approach, and cost, and their relevant experience and past performance. This information is required in the business and technical proposals. (This section discusses only the technical proposal.) The solicitation also gives the evaluation criteria that the source evaluation board will use. This section corresponds one-for-one to the items requested in the proposal instructions section.

Instructions should be stated clearly and correctly. The goal is to obtain enough information to have common grounds for evaluation. The challenge becomes how much information to give the offerors. If you are too prescriptive, the proposals may look too similar. It is important not to level the playing field too much; otherwise, discriminating among offerors will be difficult. Because the technical merits of a proposal compete with nontechnical items of similar importance (e.g., cost), the technical team should choose discriminators wisely to facilitate the source selection.

Source Evaluation Board

One or more members of the technical team serve as members of the source evaluation board. They participate in the evaluation of proposals following applicable NASA and Center source selection procedures. Because source selection is so important, the procurement office works closely with the source evaluation board to ensure that the source selection process is properly executed. The source evaluation board develops a source evaluation plan that describes the evaluation factors and the method of evaluating the offerors' responses. Source selection decisions must be carefully managed in accordance with regulations governing the fairness of the selection process.

The source evaluation board evaluates nontechnical (business) and technical items. Items may be evaluated by themselves, or in the context of other technical or nontechnical items. Table 7.1-3 shows technical items to request from offerors and the evaluation criteria with which they correlate.

Table 7.1-3 Proposal Evaluation Criteria

Item	Criteria
Preliminary contractor SEMP.	How well the plan can be implemented given the resources, processes, and controls stated. Look at completeness (how well it covers all SOW requirements), internal consistency, and consistency with other proposal items. The SEMP should cover all resources and disciplines needed to meet product requirements, etc.
Process descriptions, including subcontractor's (or team member's) processes.	Effectiveness of processes and compatibility of contractor and subcontractor processes (e.g., responsibilities, decision making, problem resolution, reporting).
Artifacts (documents) of relevant work completed. Such documentation depicts the probable quality of work products an offeror will provide on your contract. Artifacts provide evidence (or lack) of systems engineering process capability.	Completeness of artifacts, consistency among artifacts on a given project, consistency of artifacts across projects, conformance to standards.
Engineering methods and tools.	Effectiveness of the methods and tools.
Process and product metrics.	How well the offeror measures performance of its processes and quality of its products.
Preliminary subcontract management plan (may be part of contractor SEMP).	Effectiveness of subcontract monitoring and control and integration/separation of risk management and CM.
Phase-in plan (may be part of contractor SEMP).	How well the plan can be implemented given the existing workload of resources.

7.1.3.5.1 Evaluation Considerations

The following are important to consider when evaluating proposals:

- Give adequate weight to evaluating the capability of disciplines that could cause mission failure (e.g., hardware, software, thermal, optics, electrical, mechanical).
- Conduct a pre-award site visit of production/test facilities that are critical to mission success.

- Distinguish between "pretenders" (good proposal writers) and "contenders" (good performing organizations). Pay special attention to how process descriptions match relevant experience and past performance. While good proposals can indicate good future performance, lesser quality proposals usually predict lesser quality future work products and deliverables.
- Assess the contractor's SEMP and other items submitted with the proposal based on evaluation criteria that include quality characteristics (e.g., complete, unambiguous, consistent, verifiable, and traceable).
- Assess the contractor's attention to controlling program/project life-cycle costs and to placing attention on appropriate NASA/contractor surveillance, especially when NASA intends to perform the integration of contracted segments of the life cycle into a whole.

The cost estimate that the technical team performs as part of the Technical Planning Process supports evaluation of the offerors' cost proposals, helping the source evaluation board determine the realism of the offerors' technical proposals. (See Section 6.1.) The source evaluation board can determine "whether the estimated proposed cost elements are realistic for the work to be performed; reflect a clear understanding of the requirements; and are consistent with the unique methods of performance and materials described in the offeror's technical proposal."[1]

7.1.3.6 Selection of COTS Products

When COTS products are given as part of the technical solution in a proposal, it is imperative that the selection of a particular product be evaluated and documented by applying the Decision Analysis Process. Bypassing this task or neglecting to document the evaluation sufficiently could lead to a situation where NASA cannot support its position in the event of a vendor protest.

7.1.3.7 Acquisition-Unique Risks

Table 7.1-4 identifies a few risks that are unique to acquisition along with ways to manage them from an engineering perspective. Bear in mind, legal and procurement aspects of these risks are generally covered in contract clauses.

There may also be other acquisition risks not listed in Table 7.1-4. All acquisition risks should be identified and handled the same as other project risks using the Continuous Risk Management (CRM) process. A project can also choose to separate out acquisition risks as a risk-list subset and handle them using the risk-based acquisition management process if so desired.

When the technical team completes the activities prior to contract award, they will have an updated project SEMP, the Government cost estimate, an SOW, and a preliminary surveillance plan. Once the contract is awarded, the technical team begins technical oversight.

---

[1] FAR 15.404-1(d) (1).

Table 7.1-4 Risks in Acquisition

Risk	Mitigation
Supplier goes bankrupt prior to delivery	The source selection process is the strongest weapon. Select a supplier with a proven track record, solid financial position, and stable workforce. As a last resort, the Government may take possession of any materials, equipment, and facilities on the work site necessary for completing the work in-house or via another contract.
Supplier acquired by another supplier with different policies	Determine differences between policies before and after the acquisition. If there is a critical difference, then consult with the procurement and legal offices. Meet with the supplier and determine if the original policy will be honored at no additional cost. If the supplier balks, then follow the advice from legal.
Deliverables include software to be developed	Include an experienced software manager on the technical team. Monitor the contractor's adherence to software development processes. Discuss software progress, issues, and quality at technical interchange meetings.
Deliverables include COTS products (especially software)	Understand the quality of the product: Look at test results. When test results show a lot of rework to correct defects, then users will probably find more defects. Examine problem reports. These show whether or not users are finding defects after release. Evaluate user documentation. Look at product support.
Products depend on results from models or simulations	Establish the credibility and uncertainty of results. Determine depth and breadth of practices used in verification and validation of the model or simulation. Understand the quality of software upon which the model or simulation is built. For more information, refer to *NASA-STD-7009, Standard for Models and Simulations*.
Budget changes prior to delivery of all products (and contract was written without interim deliverables)	Options include: Remove deliverables or services from the contract scope in order to obtain key products. Relax the schedule in exchange for reduced cost. Accept deliverables "as is." To avoid this situation, include electronic access to data, work products, and interim deliverables to assess contractor progress on final deliverables in the SOW.
Contractor is a specialty supplier with no experience in a particular engineering discipline; for example, the contractor produces cryogenic systems that use alarm monitoring software from another supplier, but the contractor does not have software expertise	Mitigate risks of COTS product deliverables as discussed earlier. If the contract is for delivery of a modified COTS product or custom product, then include provisions in the SOW to cover the following: Supplier support (beyond product warranty) that includes subsupplier support Version upgrade/replacement plans Surveillance of subsupplier If the product is inexpensive, simply purchasing spares may be more cost effective than adding surveillance requirements.

7.1.4 During Contract Performance

7.1.4.1 Performing Technical Surveillance

Surveillance of a contractor's activities and/or documentation is performed to demonstrate fiscal responsibility, ensure crew safety and mission success, and determine award fees for extraordinary (or penalty fees for substandard) contract execution. Prior to or outside of a contract award, a less formal agreement may be made for the Government to be provided with information for a trade study or engineering evaluation. Upon contract award, it may become necessary to monitor the contractor's adherence to contractual requirements more formally. (For a greater understanding of surveillance requirements, see NPR 8735.2, Management of Government Quality Assurance Functions for NASA Contracts.)

Under the authority of the contracting officer, the technical team performs technical surveillance as established in NASA's project SEMP. The technical team assesses technical work productivity, evaluates product quality, and conducts technical reviews of the contractor. (Refer to the Technical Assessment Process.) Some of the key activities are discussed below.

- **Develop NASA-Contractor Technical Relationship:** At the contract kick-off meeting, set expectations for technical excellence throughout the execution of the contract. Highlight the requirements in the contract SOW that are the most important. Discuss the quality of work and products to be delivered against the technical requirements. Mutually agree on the format of the technical reviews and how to resolve misunderstandings, oversights, and errors.
- **Conduct Technical Interchange Meetings:** Start early in the contract period and meet periodically with the contractor (and subcontractors) to confirm that the contractor has a correct and complete understanding of the requirements and operational concepts. Establish day-to-day NASA-contractor technical communications.
- **Control and Manage Requirements**: Almost inevitably, new or evolving requirements will affect a project. When changes become necessary, the technical team needs to control and manage changes and additions to requirements proposed by either NASA or the contractor. (See Section 6.2.) Communicate changes to any project participants that the changes will affect. Any changes in requirements that affect contract cost, schedule, or performance should be conveyed to the contractor through a formal contract change. Consult the contracting officer's technical representative.
- **Evaluate Systems Engineering Processes:** Evaluate the effectiveness of defined systems engineering processes. Conduct audits and reviews of the processes. Identify process deficiencies and offer assistance with process improvement.
- **Evaluate Work Products:** Evaluate interim plans, reports, specifications, drawings, processes, procedures, and similar artifacts that are created during the systems engineering effort.
- **Monitor Contractor Performance Against Key Metrics:** Monitoring contractor performance extends beyond programmatic metrics to process and product metrics. (See Section 6.7.2.6.2 on technical performance measures.) These metrics depend on acceptable product quality. For example, "50 percent of design drawings completed" is misleading if

most of them have defects (e.g., incorrect, incomplete, inconsistent). The amount of work to correct the drawings affects cost and schedule. It is useful to examine reports that show the amount of contractor time invested in product inspection and review.

- **Conduct Technical Reviews:** Assess contractor progress and performance against requirements through technical reviews. (See Section 6.7.2.3.)
- **Verify and Validate Products:** Verify and validate the functionality and performance of products before delivery and prior to integration with other system products. To ensure that a product is ready for system integration or to enable further system development, perform verification and validation as early as practical. (See Sections 5.3 and 5.4.)

7.1.4.2 Evaluating Work Products

Work products and deliverables share common attributes that can be used to assess quality. Additionally, relationships among work products and deliverables can be used to assess quality. Some key attributes that help determine quality of work products are listed below:

- Satisfies content and format requirements,
- Understandable,
- Complete,
- Consistent (internally and externally) including terminology (an item is called the same thing throughout the documents, and
- Traceable.

Table 7.1-5 shows some typical work products from the contractor and key attributes with respect to other documents that can be used as evaluation criteria.

Table 7.1‑5 Typical Work Product Documents

Work Product	Evaluation Criteria
SEMP	Describes activities and products required in the SOW. The SEMP is not complete unless it describes (or references) how each activity and product in the SOW will be accomplished.
Software management/ development plan	Consistent with the SEMP and related project plans. Describes how each software-related activity and product in the SOW will be accomplished. Development approach is feasible.
System design	Covers the technical requirements and operational concepts. System can be implemented.
Software design	Covers the technical requirements and operational concepts. Consistent with hardware design. System can be implemented.
Human Systems Integration (HSI) Plan (if not incorporated in the SEMP)	The HSI Plan defines how human system considerations are integrated into the full systems engineering design, verification, and validation life cycle. The HSI Plan is a living document that also captures human systems issues, risks, and mitigation plans as they arise and are worked.
Installation plans	Covers all user site installation activities required in the SOW. Presents a sound approach. Shows consistency with the SEMP and related project plans.

Work Product	Evaluation Criteria
Test plans	Covers qualification requirements in the SOW. Covers technical requirements. Approach is feasible.
Test procedures	Test cases are traceable to technical requirements.
Transition plans	Describes all transition activities required in the SOW. Shows consistency with the SEMP and related project plans.
User documentation	Sufficiently and accurately describes installation, operation, or maintenance (depending on the document) for the target audience.
Drawings and documents (general)	Comply with content and format requirements specified in the SOW.

7.1.4.3 Issues with Contract-Subcontract Arrangements

In the ideal world, a contractor manages its subcontractors, each subcontract contains all the right requirements, and resources are adequate. In the real world, the technical team deals with contractors and subcontractors that are motivated by profit, (sub) contracts with missing or faulty requirements, and resources that are consumed more quickly than expected. These and other factors cause or influence two key issues in subcontracting:

1. Limited or no oversight of subcontractors, and
2. Limited access to or inability to obtain subcontractor data.

These issues are exacerbated when they apply to second-tier (or lower) subcontractors. Table 7.1-6 looks at these issues more closely along with potential resolutions.

Scenarios other than those above are possible. Resolutions might include reducing contract scope or deliverables in lieu of cost increases or sharing information technology in order to obtain data. Even with the adequate flowdown requirements in (sub) contracts, legal wrangling may be necessary to entice contractors to satisfy the conditions of their (sub) contracts.

Activities during contract performance will generate an updated surveillance plan, minutes documenting meetings, change requests, and contract change orders. Processes will be assessed, deliverables and work products evaluated, and results reviewed.

Table 7.1-6 Contract-Subcontract Issues

Issue	Resolution
Oversight of subcontractor is limited because requirement(s) missing from contract	The technical team gives the SOW requirement(s) to the contracting officer who adds the requirement(s) to the contract and negotiates the change order, including additional costs to NASA. The contractor then adds the requirement(s) to the subcontract and negotiates the change order with the subcontractor. If the technical team explicitly wants to perform oversight, then the SOW should indicate what the contractor, its subcontractors, and team members are required to do and provide.

Issue	Resolution
Oversight of subcontractor is limited because requirement(s) not flowed down from contractor to subcontractor	It is the contractor's responsibility to satisfy the requirements of the contract. If the contract includes provisions to flow down requirements to subcontractors, then the technical team can request the contracting officer to direct the contractor to execute the provisions. The contractor may need to add requirements and negotiate cost changes with the subcontractor. If NASA has a cost-plus contract, then expect the contractor to bill NASA for any additional costs incurred. If NASA has a fixed-price contract, then the contractor will absorb the additional costs or renegotiate cost changes with NASA. If the contract does not explicitly include requirements flow-down provisions, the contractor is responsible for performing oversight.
Oversight of second-tier subcontractor is limited because requirement(s) not flowed down from subcontractor to second-tier subcontractor	This is similar to the previous case but more complicated. Assume that the contractor flowed down requirements to its subcontractor, but the subcontractor did not flow down requirements to the second-tier subcontractor. If the subcontract includes provisions to flow down requirements to lower tier subcontractors, then the technical team can request the contracting officer to direct the contractor to ensure that subcontractors execute the flowdown provisions to their subcontractors. If the subcontract does not explicitly include requirements flowdown provisions, the subcontractor is responsible for performing oversight of lower-tier subcontractors.
Access to subcontractor data is limited or not provided because providing the data is not required in the contract	The technical team gives the SOW requirement(s) to the contracting officer who adds the requirement(s) to the contract and negotiates the change order, including additional costs to NASA. The contractor then adds the requirement(s) to the subcontract and negotiates the change order with the subcontractor. If the technical team explicitly wants direct access to subcontractor data, then the SOW should indicate what the contractor, its subcontractors, and team members are required to do and provide.
Access to subcontractor data is limited or not provided because providing the data is not required in the subcontract	It is the contractor's responsibility to obtain data (and data rights) necessary to satisfy the conditions of its contract, including data from subcontractors. If the technical team needs direct access to subcontractor data, then follow the previous case to add flowdown provisions to the contract so that the contractor will add requirements to the subcontract.

7.1.5 Contract Completion

The contract comes to completion with the delivery of the contracted products, services, or systems and their enabling products or systems. Along with the product, as-built documentation should be delivered and operational instructions including user manuals.

7.1.5.1 Acceptance of Final Deliverables

Throughout the contract period, the technical team reviews and accepts various work products and interim deliverables identified in the contract data requirements list and schedule of deliverables. The technical team also participates in milestone reviews to finalize acceptance of deliverables. At the end of the contract, the technical team ensures that each technical deliverable is received and that its respective acceptance criteria are satisfied.

The technical team records the acceptance of deliverables against the contract data requirements list and the schedule of deliverables. These documents serve as an inventory of items and services to be accepted. Although rejections and omissions are infrequent, the technical team needs to take action in such a case. Good data management and configuration management practices facilitate the effort.

Acceptance criteria include the following:

- Product verification and validation completed successfully. The technical team performs or oversees verification and validation of products, integration of products into systems, and system verification and validation.
- Technical data package is current (as-built) and complete.
- Transfer of certifications, spare parts, warranties, etc., is complete.
- Transfer of software products, licenses, data rights, intellectual property rights, etc., is complete.
- Transfer of maintenance, logistics, and training documentation as required.
- Technical documentation required in contract clauses is complete (e.g., new technology reports).

When the deliverable of a contract is a product and NASA has planned to manage the integration of that product into operations, it is important for NASA personnel and facilities to be ready to receive final deliverables. Key items to have planned for and prepared include the following:

- A plan for support and to transition products to operations;
- A plan for facilities support;
- A plan for logistics to support operations and maintenance of the product throughout its life cycle;
- An ongoing human systems integration plan for addressing the numbers and types of personnel required for operations;
- Training of personnel;
- Configuration management system in place; and
- Allocation of responsibilities for troubleshooting, repair, and maintenance.

7.1.5.2 Transition Management

When a contract is issued for a product but not the product's operations phase of the life cycle, before the contract was awarded, a product support strategy should have been developed as part of the life-cycle acquisition strategy. The product support strategy outlines preliminary notions regarding integration, operations, maintenance, improvements, decommissioning, and disposal. Later, after the contract is awarded, a high-level transition plan that expands the product support strategy is recorded in an appropriate document. Details of product/system transition are subsequently documented in one or more transition plans. Elements of transition planning are discussed in Section 5.5.

Transition plans should clearly indicate responsibility for each action (NASA, product contractor, or follow-on operations contractor). Also, the contract SOW should have included a requirement that the contractor will execute responsibilities assigned in the transition plan (usually on a cost-reimbursable basis).

Frequently, NASA (or NASA jointly with a prime contractor) is the system integrator on a project. In this situation, multiple contractors (or subcontractors) will execute their respective transition plans. NASA is responsible for developing and managing a system integration plan that incorporates inputs from each transition plan. The provisions that were written in the SOW months or years earlier accommodate the transfer of products and systems from the contractors to NASA.

The more detail placed in upfront documentation—especially the SOWs of the various component and service contractors—the more likely the success of overall life-cycle integration without escalation of costs due to unanticipated integration issues. Note that this attention to pre-planning is at the heart of the intent of the SEMP; i.e., that all elements (including human elements) of the system and all phases of the life cycle are considered before moving from concept through detailed design and development.

7.1.5.3 Transition to Operations and Support

When contracted or planned for execution under separate managements, the successful transition of systems to operations and support, which includes maintenance and improvements, depends on clear transition criteria that the stakeholders agree on. NASA technical and management teams should participate in the transition, providing continuity for the customers, especially when a follow-on contract is involved. When the existing system development contract is used for transition to operations, NASA technical and program management teams conduct a formal transition meeting with the contractor. Alternatively, the transition may involve the same contractor under a different contract arrangement (e.g., modified or new contract). Or the transition may involve a different contractor than the developer, using a different contract arrangement.

The key benefits of using the existing contract are that the relevant stakeholders are familiar with the contractor and the contractor knows the products and systems involved. It is important that the contractor and other key stakeholders understand the service provisions (requirements) of the contract. The formal transition meeting may lead to contract modifications in order to amend or remove service requirements that have been affected by contract changes over the years.

Seeking to retain the development contractor under a different contract can be beneficial. Although it takes time and resources to compete the contract, it permits NASA to evaluate the contractor and other offerors against operations and support requirements only. The incumbent contractor has personnel with development knowledge of the products and systems, while service providers specialize in optimizing cost and availability of services. In the end, the incumbent may be retained under a contract that focuses on current needs (not several years ago), or else a motivated service provider will work hard to understand how to operate and maintain the systems.

If a follow-on contract will be used, consult the local procurement office and exercise the steps that were used to obtain the development contract. Assume that the amount of calendar time needed to award a follow-on contract will be comparable to the time needed to award the development contract. Also consider that the incumbent may be less motivated upon losing the competition. Some items to consider for operations contracts during the development of SOW requirements include the following:

- Staff qualifications;
- Operation schedules, shifts, and staffing levels;
- Maintenance profile (e.g., preventive, predictive, run-to-fail);
- Maintenance and improvement opportunities (e.g., schedule, turnaround time);
- Historical data for similar efforts; and
- Performance-based work.

The transition to operations and support represents a shift from the delivery of products to the delivery of services. This transition should have been a part of the program/project's earliest conception and through careful systems engineering and program/project management, this transition should contain as few surprises as possible. Successful programs and projects—particularly those with successful operations and little escalation in cost growth—have conceived of the program/project as a series of elements brought together over the course of acquisition and a full life cycle to achieve planned stakeholder objectives. The use of multiple contracting mechanisms to achieve the final results can be both complicating and enabling. At no time should the program/project's managers or systems engineers lose sight of the desired end results—operational performance and life-cycle cost containment—in the face of process complexity.

Note that service-only contracts focus on the contractor's performance of activities, rather than development of tangible products. Consequently, service contract systems engineering and HSI performance standards may be more reflective of customer satisfaction and operations efficiency than those of development contracts. For example:

- Customer satisfaction ratings;
- Efficiency of service;
- Response time to a customer request;
- Availability (e.g., of system, Web site, facility);
- Time to perform maintenance action;
- Planned versus actual staffing levels;
- Planned versus actual cost;
- Effort and cost per individual service action; and
- Percent decrease in effort and cost per individual service action.

For more examples of standards to assess service contractors' performance, refer to *System and Software Metrics for Performance-Based Contracting*.

7.1.5.4 Decommissioning and Disposal

Contracts offer a means to achieve the safe and efficient decommissioning and disposal of systems and products that require specialized support systems, facilities, and trained personnel, especially when hazardous materials are involved. Consider these needs during development of the acquisition strategy and solidify them before the final design phase. Determine how many contracts will be needed across the product's life cycle.

The following are some items to consider for decommissioning and disposal during the development of SOW requirements:

- Handling and disposal of waste generated during the fabrication and assembly of the product.
- Reuse and recycling of materials to minimize the disposal and transformation of materials.
- Handling and disposal of materials used in the product's operations.
- End-of-life decommissioning and disposal of the product.
- Cost and schedule to decommission and dispose of the product, waste, and unwanted materials.
- Metrics to measure decommissioning and disposal of the product.
- Metrics to assess the contractor's performance. (Refer to *System and Software Metrics for Performance-Based Contracting*.)

7.1.5.5 Final Evaluation of Contractor Performance

In preparation for closing out a contract, the technical team gives input to the procurement office regarding the contractor's final performance evaluation. Although the technical team has performed periodic contractor performance evaluations, the final evaluation offers a means to document good and bad performance that continued throughout the contract. Since the evaluation is retained in a database, it can be used as relevant experience and past performance input during a future source selection process.

This phase of oversight is complete with the closeout or modification of the existing contract, award of the follow-on contract, and an operational system. Oversight continues with follow-contract activities.

7.2 Concurrent Engineering Methods

7.2.1 Introduction

Concurrent Engineering (CE) design techniques are an especially effective and efficient method of generating a rapid articulation of concepts, architectures, and requirements. CE is a systematic approach by diverse specialists collaborating simultaneously in a shared environment, real or virtual, to yield an integrated design. This approach is intended to cause the developers, from the very outset, to consider all elements of the product life cycle from conception to disposal while integrating cost, schedule, quality, risk, and user requirements. One of the main objectives of CE is to reduce the product development cycle time through a better integration of activities and processes.

The CE approach provides an infrastructure for brainstorming and circulating ideas between the engineers and stakeholder team representatives, which routinely results in a high-quality product that directly maps to the customer needs. The collaboration design paradigm is so successful because it enables a radical reduction in decision latency. In a non-CE environment, questions, issues, or problems may take several days to resolve. If a design needs to be changed or a requirement reevaluated, significant time may pass before all engineering team members get the information or stakeholder team members can discuss potential requirement changes. These delays introduce the possibility, following initial evaluation, of another round of questions, issues, and changes to design and requirements, adding further delays.

The tools, data, and supporting information technology infrastructure within CE provide an integrated support environment that can be immediately utilized by the team. The necessary skills and experience are gathered and are resident in the environment to synchronously complete the design. In a collaborative environment, questions can be answered immediately, or key participants can explore assumptions and alternatives with the stakeholder team or other design team members and quickly reorient the whole team when a design change occurs. The collaboration triggers the creativity of the engineers and helps them close the loop and rapidly converge on their ideas. Since the mid-1990s, the CE approach has been successfully used at several NASA Centers as well as at commercial enterprises to dramatically reduce design development time and costs when compared to traditional methods.

Although CE at NASA is based on common philosophy and characteristics, specific CE implementations vary in many areas. These variations include the following:

- The specific areas of expertise,
- The level of engineering details entertained during the study sessions,
- The type of facilitation,
- The roles and responsibilities within the CE teams as well as across CE facilities,
- Institutions and the stakeholder teams,
- The activity execution approach and the duration of study sessions,
- The configuration and attributes of the information infrastructure and knowledge base used,

- The administrative and engineering tools used, the engineering staffing approach.

Within NASA, CE is primarily used to support early project life-cycle phases, such as pre-Formulation and Formulation. In other industries, the CE process has demonstrated applicability across the full project life cycle.

7.2.2 CE Purpose and Benefits

CE stakeholders include NASA programs and projects, scientists and technologists, as well as other Government agencies (civil and military), Federal laboratories, and universities. CE products and services include the following:

- Generating mission concepts in support of Center proposals to science Announcements of Opportunity (AOs);
- Trade space exploration and architecting of systems, missions, and systems of systems;
- Full end-to-end designs, including concepts, requirements, and tradeoffs for systems and subsystems;
- Focused efforts assessing specific architecture subelements and associated tradeoffs;
- Independent assessments of customer-provided reports, concepts, and costs;
- Road mapping support; and
- Technology and risk assessments.

The principal driving forces behind the use of NASA's CE environments are increased systems engineering efficiency and effectiveness. More specifically:

- Generating more conceptual design studies at reduced cost and schedule;
- Creating a reusable process within dedicated facilities using well-defined tools;
- Developing a database of mission requirements and designs for future use; and
- Infusing a broader systems engineering perspective across the organization.

Additional resulting strategic benefits across NASA include the following:

- Core competency support (e.g. developing systems engineers, maturing and broadening of the general engineering workforce, developing mission generalists from a pool of experienced discipline engineers, providing a training environment, etc.);
- Sensitizing the customer base to cross-systems and end-to-end issues and implications of the requirements upon the design;
- Serving as a testbed for improved tools and processes;
- Providing an environment/forum for forming partnerships;
- Improving the quality and consistency of the conceptual design products; and
- Creating an environment that enables cooperative, rather than competitive, efforts among NASA organizations.

7.2.3 History of Concurrent Engineering

Historically, aerospace conceptual design studies would typically take six to eight months of time and hundreds of thousands of dollars or more to perform trade studies and arrive at a well-documented convergent point design. As a response to tightening national budgets and the resulting challenge to the Agency to create new methods to do NASA's work "faster, better, cheaper," concurrent engineering was first applied to space science mission concepts at the Jet Propulsion Laboratory (JPL) in 1995. These ideas were influenced by collaborative engineering practices from W. Edwards Deming, Total Quality Management (TQM), and other industry practices. (See Winner 1988 and Pennell 1989 for a summary of various successful CE implementations in the 1980s.)

In the aerospace implementation pioneered at JPL, CE meant co-locating scientists and engineers representing major spacecraft subsystems and disciplines, and working through the design issues of a flight project concept collaboratively and in real time, targeted at proposal support. By bringing all the requisite expertise into the same room (experts with their analysis tools and data) and by working design issues as a team, CE overcame many of the bottlenecks and communication pitfalls of the traditional design approach (sometimes called "stovepipe" design) that relied on a physically distributed team, ad hoc information transfer, action items, and periodic status meetings. As a result, CE reduced the time and cost of conceptual design drastically, such that conceptual designs can now be completed in a fraction of the traditional investment. Some authors report a reduction in cost by as much as a factor of five. (See Oberto 2005.) The subsequent rapid adoption of CE throughout the aerospace industry and its continued growth attests to its value as a design methodology.

Today, concurrent engineering is no longer an experiment or novelty; for many NASA Centers and other aerospace organizations, it is a standard concept design approach fully integrated into the organization's formulation support processes. Team X at the Jet Propulsion Laboratory (see Wall 2000 and Kluger 2005), the Integrated Design Center at the Goddard Space Flight Center (see Karpati 2003), COMPASS at the Glenn Research Center (GRC) (see McGuire 2011), the Advanced Concepts Office at the Marshall Space Flight Center (MSFC) (see Mulqueen), the Concept Design Center team at the Aerospace Corporation, and the Concurrent Design Facility at the European Space Research and Technology Center (ESTEC) are only a few examples of the CE teams currently operating. The community has grown to include industrial and academic organizations. Many university engineering programs now include coursework on CE. Students are at times invited to participate in NASA CE teams as part of student projects in several concurrent engineering facilities. New teams that look beyond the traditional point design focus of CE studies, primarily by enabling concept generation or architecture studies, are becoming operational, addressing the need for a broader range of more versatile CE services. For example, the Rapid Mission Architecture (RMA) team developed in 2007 (see Moeller 2011) and the A-Team developed in 2009 (see Ziemer 2013) at JPL, and the Architecture Design Laboratory at Goddard developed in 2010 are relatively new teams that look at architecture formulation and trade space exploration for systems and systems of systems.

Over the past decade, the concurrent engineering teams at different aerospace organizations have evolved largely independently. The different teams conduct studies using different processes, with some teams doing virtually all of the design work in real-time concurrent sessions, and

others doing more work outside the sessions. However, with a growing need for collaboration between the NASA Centers as well as industry partners and international space agencies – due to reduced budgets and an anticipated increasing number of multi-Center and multi-Agency missions – it is likely that the CE teams will need to interact more often than they have in the past. This will require a significant change in the current state of practice to enable effective electronic and real time interfaces.

One useful way of conceptualizing the different design team needs is by looking at the maturity of the concepts that they assess. Concept Maturity Level (CML) is a recently created measure for assessing the maturity of an evolving concept. (See Warfield 2010 and Wessen 2013.) The rating scale is presented in Table 7.2-1. Similar to the notion of NASA Technology Readiness Levels (TRLs), which reflect key points along the technology maturation pathway and their associated technology development characteristics, the idea for a CML scale is to address the common path of progression through the mission formulation phase from initial idea through Critical Design Review (CDR). Varying levels of concept maturity may entail differing levels of fidelity of engineering analysis, broader versus more localized trades, or varying techniques for cost and schedule analysis. For example, JPL's Team X and GRC's COMPASS are CML 4 teams, while Goddard's Architecture Design Laboratory and JPL's A-Team are CML 2/3 facilities. A NASA project mostly works from CML 5 to CML 9. For additional information on CML, see Section 8.3. Concept maturity levels are defined in Table 7.2-1.

Table 7.2-1 Concept Maturity Levels

CML	Name	Description
1	Cocktail napkin	Objectives and basic approach.
2	Initial feasibility	High-level physics, mass, and cost assessments. Validate that the mission (or instrument) concept is viable.
3	Trade space	Expansion of objectives and architecture trade space with elaboration and evaluation of performance, cost, and risks.
4	Point design within trade space	Subsystem-level design and cost estimates.
5	Concept baseline	Relationships and dependencies, partnering, heritage, technologies, key risks, mitigation plans, and system make-buy approaches
6	Initial design	Requirements and schedules to subsystem level, grassroots cost agreements, schedule, and V&V approach for key areas.
7	Preliminary cost-schedule-design integrated baseline	PMSR/MDR; preliminary project plan.
8	Final cost-schedule-design integrated baseline	PDR; baseline project plan.
9	Detailed system design	CDR

7.2.4 Key Elements of a Successful Concurrent Engineering Team

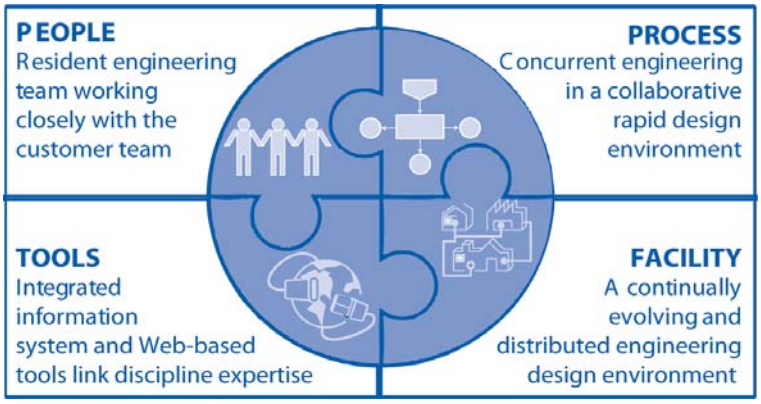


Figure 7.2-1 CE People/Process/Tools/Facility Paradigm

NASA CE is built upon a people/process/tools/facility paradigm that enables the accelerated production of high-quality engineering design concepts in a concurrent, collaborative, and rapid design environment. (See Figure 7.2-1.) The CE environment typically involves the collocation of an in-place leadership team and core multidisciplinary engineering team working with a stakeholder team using well-defined processes in a dedicated collaborative, concurrent engineering facility with specialized tools. The engineering and collaboration tools are connected by the facility's integrated infrastructure. The teams work synchronously for a short period of time in a technologically intensive physical environment to complete a design. CE facilities are most often used to design space instruments and payloads or missions including orbital configuration; hardware such as spacecraft, landers, rovers, probes, or launchers; data and ground communication systems; other ground systems; and mission operations. But the CE process applies beyond narrowly defined instrument and/or mission conceptual design, and has been used successfully for systems, mission, and system of systems, architecting, as well as other endeavors. The following sections describe each of these key elements to the successful CE implementation.

7.2.4.1 People and Staffing a Concurrent Engineering Team

The success of a concurrent engineering center is primarily based on the talented and experienced group of engineers and scientists that make up the team, who are supported by the appropriate tools and facilities in order to do their job more effectively. Concurrent engineering teams typically have several key positions: Customer, study lead, systems engineers, and subsystem or discipline engineers (including risk and cost experts). In a CE environment, the engineering team directly interacts with the stakeholders to facilitate the design, and the customer becomes an active participant in the design process. As the people involved are the most important component of CE, developing a team of engineers that can work together effectively and produce high-quality products is the highest priority for any CE center. Problems related to creating such a team are currently some of the most challenging to solve for the CE teams.

A CE Team consists of a management or leadership team, a multidisciplinary engineering team, a stakeholder team, and a facility support team. These are all vital elements in achieving a successful CE activity.

- The CE operations manager serves as the CE facility's advocate and manager. He/she provides coordination with potential and actual customers from first contact through final delivery of the CE products, and arranges/negotiates scheduling and funding for the study and negotiates the products and costs. His primary responsibility is to maintain and evolve the operational CE capability by infusing continuous process and product improvements as well as evolutionary growth into it to ensure the CE environment's continued relevance.
- A capable CE team lead, who is typically also the study facilitator, is crucial for success. The team lead coordinates and facilitates the team's concurrent study activity, and is the primary interface to the stakeholders aiming to ensure that the customer objectives are adequately captured and represented in the design. The team lead maintains overall situational awareness in the rapid fire CE environment. The team lead makes sure all team members stay involved and are effectively communicating with the other team members who are critical to completing their particular portion of the design.
- The engine of every CE team is a cadre of experienced engineers, each representing a different discipline or crosscutting engineering domain. The team of discipline engineers is headed by a lead systems engineer who works hand in hand with the team lead or facilitator. The core engineering team may be supplemented with additional specialty (crosscutting) and/or nonstandard engineering experts as required to meet any unique stakeholder need. These supplementary engineering capabilities can be obtained either from the local Center or from an external source. All engineers on the team are equipped with the techniques and software tools regularly used in their areas of expertise, and all interact with the team lead, the lead systems engineer, the other engineers on CE the team, and the stakeholders to study the feasibility of proposed solutions and to produce the final design for their specific subsystem.
- A CE facility support team maintains and develops the physical and information infrastructure to support CE activities.

Critical Issues: Getting and maintaining the best staff for the team and how to maximize efficient collaboration

7.2.4.1.1 Getting and Maintaining the Best Staff for the Team

Not all engineers work effectively in a concurrent engineering environment. Engineers who are successful in CE teams are generally comfortable with working with many unknowns and can easily adapt to rapid changes. They are able to work as part of a team and communicate effectively with stakeholders. Study leads must embody these qualities as well, augmented by leadership and systems engineering skills and a broad experience in engineering systems. Engineers with these characteristics are difficult to find and retain, as they are sought after by flight projects as well. There is also often a cultural bias in organizations that favors implementation work over formulation work, so it is a challenge to incentivize conceptual design efforts in order to retain highly qualified engineers.

Another aspect of maintaining the staff involves understanding the ideal combination of senior and junior engineers that enables an effective CE team. In order to get the best engineers for the team, it is essential that these characteristics be well managed and supported.

7.2.4.1.2 Maximizing Efficient Collaboration

Converting a set of individuals into a cohesive, high performing team is not an easy task. Effective teams can be difficult to establish and maintain, especially if the composition of the team in a given session is different each time due to the use of a rotating cast of experts. While it is expected that the individuals will be competent in their own particular fields, they need training in specific aspects of the concurrent environment. The episodic nature of collaborative design studies and other project commitments of the engineers limit the duration and quality of time available for team-building. While some funds are available for tangible products like tools and products in CE teams, there is often limited funding for training the people involved in order to improve collaboration. Excessive turnover in the team undermines the efficiency of collaboration within the team. Hence preserving the optimal balance of turnover and stability in staffing is important for maintaining a high performing team. An effective team does not just materialize by itself; it is purposefully built. The most appealing organizational vision and efficient processes and tools will not be achieved or used effectively if the team does not have the right expertise or work well together. Traditional team-building activities should be budgeted for explicitly on an equal, if not greater, priority level than process and tool improvements.

7.2.4.2 The Concurrent Engineering Process

The primary goal of the CE process is to ensure that the study meets the customer requirements in an effective manner within the time and cost allocated. The process must make the best and most efficient use of the experts and their tools in creating a design. Careful planning is crucial in achieving these goals.

It is challenging to develop a process that is consistent and repeatable, but is also flexible enough to allow for changes needed during a CE session. As the members of a CE team typically vary across studies, it is important to have consistent processes in place to reduce the variation in the study output. It is not required that the process be the same across concurrent teams at different Centers, but it is necessary to be able to define the interfaces between the different teams during distributed collaborative design sessions, similar to interface agreements between subsystems in traditional projects.

A consistent step-by-step process is essential to reach a conclusion and finish a design in an allotted amount of time. The individual substeps differ in response to the needs and the makeup of the individual concurrent teams. However, the main steps are applicable to all concurrent teams. The following outline as well as the steps shown in Figure 7.2-2 capture, at the very top level, a representative process for a design sequence from the germ of an idea to the final products. The details of each of the steps may vary between CE teams, but the main steps remain the same. The amount of time taken to complete a particular step or study can vary from days to weeks to months, depending on the level of detail of the study or the complexity of the mission.

7.2.4.2.1 Establishing the Scope

In order to make the most of the design team, it is essential to start the study with a solid problem definition. The team lead/study facilitator meets with the customer to understand the problem to be solved and develop the requirements for the design session. The team lead and the customer agree on the top level requirements, figures of merit, goals of the design study, required products, study schedule, and any other engineering or study constraints. Each team has different products to offer. The products typically range from annotated presentations, CAD models, and spreadsheet summaries to a full text report. The level of effort, time to completion, and cost to the customer vary as a function of the scope and depth of detail of the desired analyses and products.

This is where the study planning and preparation activities are conducted, the stakeholder-provided data and the objectives and activity plan are reviewed, and the scope of the activity is finalized. A discussion is held of what activities need to be done by each of the stakeholders and the design teams. For example, for planning a mission design study, the customer identifies the mission objectives by defining the measurement objectives and the instrument specifications, as applicable, and identifying the top-level requirements. Due to the relatively short duration of the CE study, the preliminary work, which is required to enable the CE study but may take a long time, is performed by a subset of the CE engineering team before the start of the actual study. Typical long duration work items include flight dynamics analyses; entry, descent, and landing profiles; launch vehicle performance trajectory analyses; thrust and navigation analyses; and complex optical analyses. Those tasks must be identified in the planning meetings to enable the rapid flow of true CE in the study execution phase. The level of analysis in this phase is a function of many factors, including the level of maturity of the incoming design, the stated goals and objectives of the engineering activity, supporting engineer availabilities, and scheduling.

7.2.4.2.2 Pre-Study Background Work

The amount of background work done prior to the CE session varies by team and by the type of mission being studied. In preparation for the study, team members typically review similar previous missions and perform all necessary early work, especially on the long lead items mentioned above. They may also discuss specific aspects of the mission with the customer to gain a better understanding of the higher level mission requirements and constraints.

7.2.4.2.3 Full-Team Concurrent Design Sessions

A design session is the physical or virtual meeting during which the members of the concurrent team gather to perform the analyses and information exchanges necessary to design a system collectively. The activities and outputs of the design session depend on the type of study being conducted and the level of conceptual maturity of the mission. Different teams develop their designs on different timescales, which are also a function of the amount of work done in real-time concurrent sessions versus independent work performed outside the CE sessions. The study products may vary from high-level mission feasibility studies aiming to determine if a concept is viable, to detailed convergent point designs, some based on high-level, even parametric subsystem concepts, while others include very detailed system and subsystem designs as well as cost and schedule estimates based on detailed concept of operations and master equipment lists. During the design session, the concurrent design team works with the customer team to address

the desired level of fidelity. Ideally, designs (or a set of architectures for trade studies) are iterated either until full convergence is achieved or until they are determined to be infeasible. Convergence is usually driven by a combination of key parameters and constraints, which typically include, as a minimum, mass, power, cost, schedule, data, and launch vehicle constraints.

A typical activity or study begins with the customer presentation of the overall mission concept and instrument concepts, as applicable, to the entire team. Additional information provided by the customer / stakeholders includes the team objectives; the science and technology goals; the initial requirements for payload, spacecraft, and mission design; the task breakdown between providers of parts or functions; top challenges and concerns; and the approximate mission timeline. This information is often provided electronically in a format accessible to the engineering team, and is presented by the customer / stakeholder representatives at a high level. During this presentation, each of the discipline engineers focuses on the part of the overall design that is relevant to his or her subsystem. The systems engineer enters the high-level system requirements into the master systems spreadsheets or master database that is used throughout the CE process to track and document the evolution of the design. The data sources can be projected on large overhead displays as well as the CE team members' individual screens to keep the entire team synchronized and the customer/stakeholders aware of the latest developments.

The engineering work is performed iteratively, with the team lead and systems engineer playing key roles to lead the process. Thus, issues are quickly identified, so consensus on tradeoff decisions and requirements redefinition can be achieved while maintaining momentum. The customer team actively participates in the collaborative process (e.g., trade studies, requirements relaxation, clarifying priorities), contributing to the rapid development of an acceptable product.

Each discipline maintains a set of key parameters used to describe its design. Because of the interdependencies among the various subsystems, each discipline engineer needs to know the value of certain parameters describing other subsystems. These parameters are shared through the local CE information infrastructure network. Often, there are conflicting or competing objectives for various disciplines and tradeoffs must be conducted overarching several subsystems. Such tradeoffs are typically defined and led by the systems engineer. The physical layout of the seating arrangement is designed such that subsystems which need to interact extensively with each other in tradeoffs and other matters are clustered in close physical proximity to facilitate communication.

Sidebars and Tag-ups

Two key aspects of a design session are the sidebar and the tag-up. These are critical in maintaining the flow of information and situational awareness across all team members. At times, more in depth discussions are needed than what is possible in the main CE room. A sidebar is the means of accomplishing that. A sidebar is a break-out session in which only a subset of the team participates to discuss a particular issue related to the study. When a sidebar is initiated, the participants physically move into a side room set aside for that purpose and conduct their discussions there. After the conclusion of the sidebar, its participants usually report back on the outcome of their discussions to the whole team at the next general session.

A tag-up is an all-team activity that is used to keep the entire team, as well as the customers and stakeholders, synchronized and up to date. Tag-ups are typically held once or twice per day in the main CE room during study sessions. At the tag-up, all team members focus on the front of the room, and only a single discussion is allowed. Team member take turns round-robin style to present the status of their portion of the design and briefly discuss any issues associated with it that are of interest to the whole team. Tag-ups are the primary means in the study to maintain overall situational awareness. Tag-ups also force subsystems to adopt a systems perspective relative to their designs. Tag-ups are implemented differently across various CE teams, but their purpose is the same for all.

7.2.4.2.4 Post-Session Documentation and Presentations

While there is a large variation in the post-design session activities between teams, all teams develop a product that documents the final design using a consistent template and also present that design to the customer. Products may include PowerPoint slides, text documents, configuration drawings, trajectory files, various analysis results, and computer models, delivered both in presentations and in appropriate cyber formats.

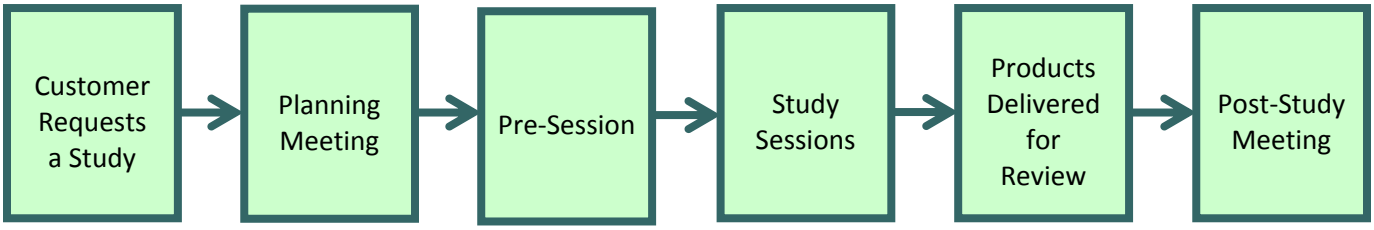


Figure 7.2-2 Concurrent Engineering Process

Critical Issue: Process integration with joint studies

A key process issue arises when conducting a joint study between multiple CE teams because each team has somewhat different core capabilities and associated processes and operates on different timescales. It is essential for each team to have a good understanding of what the capabilities and processes are for each of the other teams as collaboration between teams becomes a common occurrence. This also means that standardized products and a consistent process are necessary within a team to be able to create the proper interfaces with other teams. Coordinating the different process timescales between teams is a challenge. For example, if one team does most of its design work in real time, and another primarily works out of session, collaboration between the two will be difficult. Changes will need to be made to the processes of the teams to ensure compatibility during distributed design sessions. In order to identify the process changes necessary to enable collaborative design, an understanding of the current processes and capabilities of the teams is needed.

7.2.4.3 Concurrent Engineering Products

The products generated by CE teams vary greatly from basic feasibility of mission concepts to detailed point designs, depending on the CE facility as well as the customer's requirements. Over the years, it has been found that the study products can be captured most efficiently in a presentation slide format. However, most teams maintain the capability to produce a formal

report, such as was required by the 2010 Planetary Decadal Survey. Delivered products can also include model data and specifications in spreadsheets and CAD drawings/models. While there is significant similarity in the products generated by each team, there are differences due to the types of missions and customer needs, as well as the level of detail provided in the products.

As a minimum, baseline study products cover the following areas, typically for the system and also for each subsystem, both in prose and in sketches, drawings, figures, tables, spreadsheets, and other formats and media, as appropriate:

- Mission objectives;
- Design assumptions;
- Design drivers;
- Trajectory and orbital parameters;
- System and subsystem designs;
- Ground systems and networks assessments, link calculations;
- Launch vehicles evaluation;
- Integration and test assessment;
- Key components;
- High-level project schedule;
- Resource estimates: cost, mass, power, data rates, Delta-Velocity (ΔV) budget;
- Trades conducted;
- Risk assessment;
- Future work; and
- Issues and concerns.

Some teams are also capable of generating more detailed analyses, such as the following:

- Science requirements and traceability matrix;
- Mission animation;
- Flight equipment and master equipment lists;
- Concept of operations;
- Ground system design;
- Orbit determination, tracking;
- EDL details;
- Technical risk evaluation and technology needs definition;
- Integrated modeling/integrated analysis products; and
- Cost estimates (high-level similarity based or parametric, grass roots, master equipment list-based detailed parametric, or a mixture of the above).

Newer, lower CML teams also produce other products, such as the following:

- Trade-space analysis products, architecture trade matrices;
- Science value matrices; and
- High-level cost and risk analyses.

In addition to formal delivery of the CE product to the customer team, the final results and planning data are also archived within the CE environment for safekeeping, future reference, and for inclusion in internal cross-study analyses.

Critical Issue: Understand the products available from different teams

Presently, study products vary widely between design Centers in form, content, and even medium. Use of a standardized product set would enable the smooth transfer of results to customers and industry and easier collaboration between design centers, and make archiving and searching more efficient. In cases where the outputs of multiple teams need to be compared, such as in the 2010 Planetary Decadal Survey, a detailed understanding of the assumptions, inputs, and outputs is necessary. As collaborative distributed design involving multiple CE teams is becoming more common, the data products of each of the teams must be well understood to enable close collaboration during design sessions.

Critical Issue: Review of products (system and subsystem) in a timely manner by appropriate reviewers

Since the speed of the CE process differs so greatly from the normal engineering processes, an appropriate method of product review is challenging. To a certain extent, the study lead and systems engineers can evaluate the subsystem design, but it is preferable to have other subsystem engineers (outside of the CE team) review the design, especially when new technologies or techniques are being proposed.

7.2.4.4 Tools

Concurrent engineering design centers depend vitally on specialized and unique tools as essential enablers for their efficiency and productivity. Some of these tools are purchased off-the-shelf, while some are developed in-house. While a great variety of tools are deployed at major concurrent aerospace design centers, the similarities in categories and types of tools are striking. The tools can be classified according to the following taxonomy:

- **Concurrent collaboration tools**: data exchange platforms, in-lab audiovisual tools, remote presence tools;
- **Engineering tools**: system-level and tally tools, subsystem and discipline design tools (parametric sizing and estimation tools, analysis, and modeling tools);
- **Study management tools**: customer interface and data transfer tools, support personnel assignment tools;
- **Lab management tools**: IT and Web tools, procedure, administrative, procurement, and financial tools; and
- **Costing tools**: parametric and grassroots costing tools.

Engineering tools and techniques vary within and across CE environments in several technical aspects, such as the level of fidelity, level of integration, generally available commercial applications versus custom tools versus customized knowledge-based Excel spreadsheets, degree of parametric design versus engineering analysis. For example, mechanical design tools range

from whiteboard discussions to notepad translations to computer-aided design to 3D rapid prototyping.

Important factors in determining which tools are appropriate to an activity include the purpose and duration of the activity, the engineers' familiarity or preference, the expected product, the local culture, and the evolution of the engineering environment. Factors to be considered in the selection of CE tools and engineering techniques should also include flexibility, compatibility with the CE environment and process, and value and ease of use for the customer after the CE activities.

Engineering tools may be integrated into the CE infrastructure, routinely provided by the supporting engineering staff, and/or utilized only on an activity-by-activity basis, as appropriate. Also, as required, auxiliary engineering analyses outside of the scope of the CE effort can be performed external to the CE environment and imported for reference and incorporation into the CE product.

As CE teams have evolved, they have developed very specific tools that are optimized to meet a particular set of needs. Hence, those tools are often not flexible enough to be applied to concepts at other levels of maturity.

Critical Issue: CE teams need to be able to adapt to changing customer needs

In order to meet evolving customer needs and expand the applicability of CE, models at various levels of fidelity should be developed. Being able to integrate diverse tools for different CMLs from trade-space exploration tools to simulation-based models and detailed design models would allow CE teams to support conceptual design from the early architecture trade phase to point designs. Use of model integration tools, such as MBSE, that support plug-and-play of a wide variety models may allow the use of appropriate models for different scenarios rather than the one-size-fits-all tool sets in use today.

7.2.4.5 Concurrent Engineering Facilities

A CE facility has only one requirement: to support and enhance real time collaborative communication. The structure of CE rooms and the supporting equipment varies based on whether the room is intended to support detailed point design, architecture and trades, or brainstorming initial ideas. The facility setup for the point design rooms, typically concept maturity level 4 (CML 4) facilities, appears to be well understood, as all of the major design centers are configured in a very similar manner. For example, the facility configuration for JPL's design center is shown in Figure 7.2-3. There are typically two to three large screens in the front of the room for projection of the displays from multiple stations simultaneously and there can be additional screens on the sides of the rooms. There needs to be high-quality audio for participants calling in from external sites. The most critical element is that the stations for the subsystem chairs have a clear line of vision with each other, the customer, and the various screens in the room. Figure 7.2-4 shows a typical interaction during a study at the Mission Design Lab at Goddard Space Flight Center (GSFC). It is ideal if the setup includes a minimum of two support rooms, one for sidebars and break-out sessions and the other to house the servers that provide the IT infrastructure for the linked tools that are operated from each workstation as well as the various databases. Typically, every workstation is interlinked by a secure local area

network such that all data parameters can be shared and updated on all screens in real time. Internet-based intra-study links are in experimental phases as data security is a paramount issue.

The differences in processes and products used by lower CML teams require different facilities compared to CML 4 teams. The room configuration for lower CML studies (e.g., idea generation, early architecture trades) is less standardized and emphasizes the need for far greater flexibility and smaller size teams. These types of facilities are typically smaller, require equipment for effective distributed communication (clear audio), have break-out rooms/areas, and usually do not require highly capable computer hardware or assigned workstations, as CAD modeling is typically not performed by such teams.

As the applicability of CE expands to later stages of the life cycle, it is likely that different facilities will be needed for higher CML teams as well.

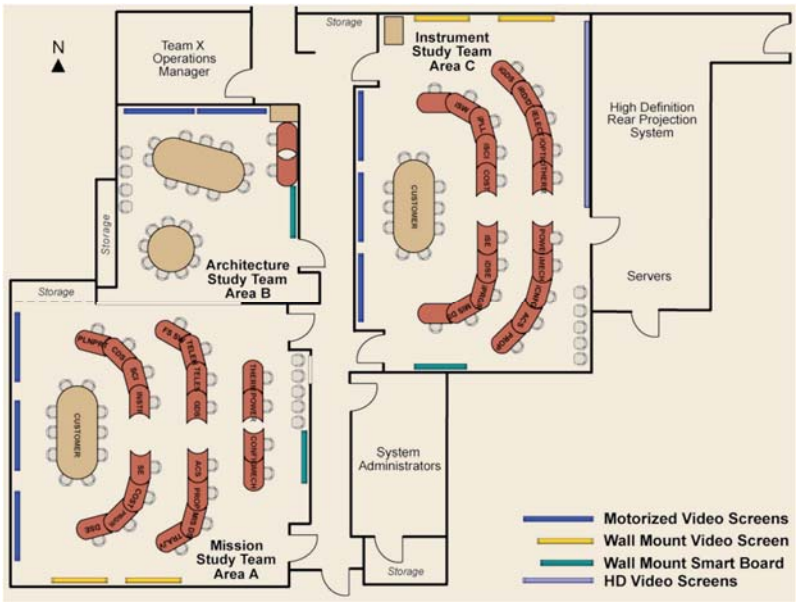


Figure 7.2-3 JPL Team X Concurrent Design Facility Configuration

Figure 7.2-4 GSFC Integrated Design Center Study Session

Critical Issues: Usability of technology

There is a tendency, especially with new teams starting up, to choose complicated and expensive display devices such as computer-linked white boards and other high-end devices. All of the established teams have found that these are rarely used because of the complications and the learning curve associated with them. In the rapid-fire exchange of ideas in a CE room, there is no time to learn how to use a new device on the fly, and these devices are not used frequently enough by the team to remember their operation. For maximum efficiency, everything in the room needs to be easy to use and often low tech but high quality tools are preferred. For example, a whiteboard camera, which costs a few hundred dollars and generates jpeg files to a computer over a wireless connection, enables the team members to use standard markers and still, with the click of a button, the image on the board can be saved for future reference.

7.3 Selecting Engineering Design Tools

NASA utilizes cutting-edge design tools and techniques to create the advanced analyses, designs, and concepts required to develop unique aerospace products, spacecraft, and science experiments. The diverse nature of the design work generated and overseen by NASA requires use of a broad spectrum of robust electronic tools such as computer-aided design tools and computer-aided systems engineering tools. Based on the distributed and varied nature of NASA projects, selection of a single suite of tools from only one vendor to accomplish all design tasks is not practical. However, opportunities to improve standardization of design policy, processes, and tools remain a focus for continuous improvement activities at all levels within the Agency.

These guidelines serve as an aid to help in the selection of appropriate tools in the design and development of aerospace products and space systems and when selecting tools that affect multiple Centers. If no tools exist or can be adapted, the option of developing a new tool should be within the options considered, ranging from internal NASA development, partnerships with industry and academia, or a dedicated procurement.

7.3.1 Program and Project Considerations

When selecting a tool to support a program or project, all of the upper-level constraints and requirements should be identified early in the process. Pertinent information from the project that affects the selection of the tools will include the urgency, schedule, resource restrictions, extenuating circumstances, and constraints. A tool that does not support meeting the program master schedule or is too costly to be bought in sufficient numbers will not satisfy the project manager's requirements. For example, a tool that requires extensive modification and training that is inconsistent with the master schedule should not be selected by the technical team. If the activity to be undertaken is an upgrade to an existing project, legacy tools and availability of trained personnel are factors to be considered.

7.3.2 Policy and Processes

When selecting a tool, it is important to consider the applicable policies and processes at all levels, including those at the Center level, within programs and projects, and at other Centers when a program or project is a collaborative effort. In the following discussion, the term "organization" will be used to represent any controlling entity that establishes policy and/or processes for the use of tools in the design or development of NASA products. In other words, "organization" can mean the user's Center, another collaborating Center, a program, a project, inline engineering groups, or any combination of these entities.

Policies and processes affect many aspects of a tool's functionality. First and foremost, there are policies that dictate how designs are to be formally or informally controlled within the organization. These policies address configuration management processes that should be followed as well as the type of data object that will be formally controlled (e.g., drawings or models). Clearly this will affect the types of tools that will be used and how their designs will be annotated and controlled.

The Information Technology (IT) policy of the organization also needs to be considered. Data security and export control (e.g., International Traffic in Arms Regulations (ITAR)) policies are

two important IT policy considerations that will influence the selection of a particular design tool.

The policy of the organization may also dictate requirements on the format of the design data that is produced by a tool. A specific format may be required for sharing information with collaborating parties. Other considerations are the organizations' quality processes, which control the versions of the software tools as well as their verification and validation. There are also policies on training and certifying users of tools supporting critical flight programs and projects. This is particularly important when the selection of a new tool results in the transition from a legacy tool to a new tool. Therefore, the quality of the training support provided by the tool vendor is an important consideration in the selection of any tool.

Also, if a tool is being procured to support a multi-Center program or project, then program policy may dictate which tool should be used by all participating Centers. If Centers are free to select their own tool in support of a multi-Center program or project, then consideration of the policies of all the other Centers should be taken into account to ensure compatibility among Centers.

7.3.3 Collaboration

The design process is highly collaborative due to the complex specialties that should interact to achieve a successful integrated design. Tools are an important part of a successful collaboration. To successfully select and integrate tools in this environment requires a clear understanding of the intended user community size, functionality required, nature of the data to be shared, and knowledge of tools to be used. These factors will dictate the number of licenses, hosting capacity, tool capabilities, IT security requirements, and training required. The sharing of common models across a broad group requires mechanisms for advancing the design in a controlled way. Effective use of data management tools can help control the collaborative design by requiring common naming conventions, markings, and design techniques to ensure compatibility among distributed design tools.

7.3.4 Design Standards

Depending on the specific domain or discipline, there may be industry and Center-specific standards that should be followed, particularly when designing hardware. This can be evident in the design of a mechanical part, where a mechanical computer-aided design package selected to model the parts should have the capability to meet specific standards, such as model accuracy, dimensioning, and tolerancing, the ability to create different geometries, and the capability to produce annotations describing how to build and inspect the part. However, these same issues should be considered regardless of the product.

7.3.5 Existing IT Architecture

As with any new tool decision, an evaluation of defined Agency and Center IT architectures should be made that focuses on compatibility with and duplication of existing tools. Typical architecture considerations would include data management tools, middleware or integration infrastructure, network transmission capacity, design analysis tools, manufacturing equipment, approved hosting, and client environments.

While initial focus is typically placed on current needs, the scalability of the tools and the supporting IT infrastructure should be addressed too. Scalability applies to both the number of users and capacity of each user to successfully use the system over time.

7.3.6 Tool Interfaces

Information interfaces are ubiquitous, occurring whenever information is exchanged.

This is particularly characteristic of any collaborative environment. It is here that inefficiencies arise, information is lost, and mistakes are made. There may be an organizational need to interface with other capabilities and/or analysis tools, and understanding the tools used by the design teams with which your team interfaces and how the outputs of your team drive other downstream design functions is critical to ensuring compatibility of data.

For computer-aided systems engineering tools, users are encouraged to select tools that are compatible with the Object Management Group (OMG) System Modeling Language (SysML) standard. SysML is a version of the Unified Modeling Language (UML) that has been specifically developed for systems engineering.[2]

7.3.7 Interoperability and Data Formats

Interoperability is an important consideration when selecting tools. The tools should represent the designs in formats that are acceptable to the end user of the data. It is important that any selected tool include associative data exchange and industry-standard data formats. As the Agency increasingly engages in multi-Center programs and projects, the need for interoperability among different tools, and different versions of the same tool, becomes even more critical. True interoperability reduces human error and the complexity of the integration task, resulting in reduced cost, increased productivity, and a quality product.

When considering all end users' needs, it is clear that interoperability becomes a difficult challenge. Three broad approaches, each with their own strengths and weaknesses, are:

1. Have all employees become proficient in a variety of different tool systems and the associated end use applications. While this provides a broad capability, it may not be practical or affordable.
2. Require interoperability among whatever tools are used, i.e., requiring that each tool be capable of transferring model data in a manner that can be easily and correctly interpreted by all the other tools. Considerable progress has been made in recent years in the standards for the exchange of model data. While this would be the ideal solution for many, standard data formats that contain the required information for all end users do not yet exist.
3. Dictate that all participating organizations use the same version of the same tool. When the use of same version of the tool is not possible, version and model controls will be necessary to validate models and simulations in differing tools.

---

[2] OMG, UML, and SysML are either registered trademarks or trademarks of Object Management Group, Inc. in the United States and/or other countries.

7.3.8 Backward Compatibility

On major programs and projects that span several years, it is often necessary to access design data that are more than 3 to 5 years old. However, access to old design data can be extremely difficult and expensive, either because tool vendors end their support or later versions of the tool can no longer read the data. Strategies for maintaining access include special contracts with vendors for longer support, archiving design data in neutral formats, continuous migration of archives into current formats, and recreating data on demand. Organizations should select the strategy that works best for them, after a careful consideration of the cost and risk.

7.3.9 Platform

While many tools will run on multiple hardware platforms, some perform better in specific environments or are only supported by specified versions of operating systems. In the case of open-source operating systems, many different varieties are available that may not fully support the intended tools. If the tool being considered requires a new platform, the additional procurement cost and administration support costs should be factored in.

7.3.10 Tool Configuration Control

Tool configuration control is a tradeoff between responsive adoption of the new capabilities in new versions and smooth operation across tool chain components. This is more difficult with heterogeneous (multiple vendor) tool components. An annual or biannual block upgrade strategy requires significant administrative effort. On the other hand, the desktop diversity resulting from user-managed upgrade timing also increases support requirements.

7.3.11 Security/Access Control

Special consideration should be given to the sensitivity and required access of all design data. Federal Government and Agency policy requires the assessment of all tools to ensure appropriate security controls are addressed to maintain the integrity of the data. The systems engineer should work with the Organizational Computer Security Officer (OCSO) to integrate IT Security into the system. Important activities include development of the security plan and the emergency response plan per NPR 7120.7 For more details on IT security, see NPR 2810.1, Security of Information Technology.

7.3.12 Training

Most of the major design tools have similar capabilities that will not be new concepts to a seasoned designer. However, each design tool utilizes different techniques to perform design functions, and each contains some unique tool sets that will require training. The more responsive vendors will provide follow-up access to instructors and onsite training with liberal distribution of training materials and worked examples. The cost and time to perform the training and time for the designer to become proficient can be significant and should be carefully factored in when making decisions on new design tools.

The disruptive aspect of training is an important consideration in adapting to a different tool. Before transitioning to a new tool, an organization should consider the schedule of deliverables

to major programs and projects. Can commitments still be met in a timely fashion? It is suggested that organizations implement a phase-in approach to a new tool, where the old tool is retained for some time to allow people to learn the new tool and become proficient in its use. The transition of a fully functional and expert team using any one system to the same team fully functional using another system is a significant undertaking. Some overlap between the old tool and the new tool will ensure flexibility in the transition and ensure that the program and project work proceeds uninterrupted.

7.3.13 Licenses

Licenses provide and control access to the various modules or components of a product or product family. Consideration of the license scheme should be taken into account while selecting a tool package. Licenses are sometimes physical, like a hardware key that plugs into a serial or parallel port, or software that may or may not require a whole infrastructure to administer. Software licenses may be floating (able to be shared on many computers on a first-come, first-served basis) or locked (dedicated to a particular computer). A well-thought-out strategy for licenses should be developed in the beginning of the tool selection process. This strategy should take into consideration program and project requirements and constraints as well as other factors such as training and use. The strategy development should involve the applicable IT organization.

7.3.14 Stability of Vendor and Customer Support

As in the selection of any support device or tool, vendor stability is of great importance. Given the significant investment in the tools (directly) and infrastructure (indirectly), it is important to look at the overall company stability to ensure the vendor will be around to support the tools. Maturity of company products, installed user base, training, and financial strength can all provide clues to the company's ability to remain in the marketplace with a viable product. In addition, a responsive vendor provides customer support in several forms. A useful venue is a Web-based user-accessible knowledge base that includes resolved issues, product documentation, manuals, white papers, and tutorials. Live telephone support can be valuable for customers who don't provide support internally. An issue resolution and escalation process involves customers directly in prioritizing and following closure of critical issues. Onsite presence by the sales team and application engineers, augmented by post-sales support engineers, can significantly shorten the time to discovery and resolution of issues and evolving needs.

7.4 Environmental, Nuclear Safety, and Planetary Protection Policy Compliance

7.4.1 National Environmental Policy Act (NEPA) and Executive Order 12114

The National Environmental Policy Act (NEPA) established the White House Council on Environmental Quality (CEQ), which published NEPA implementing regulations. The regulations require agencies to consider potential environmental effects when planning programs and projects. NASA has developed Agency-specific NEPA regulations and policies to ensure compliance with NEPA and its implementing regulations, as well as Executive Order (EO) 12114, *Environmental Effects Abroad of Major Federal Actions.*

NASA NEPA regulations (14 CFR Part 1216.3) have existed since 1979. The regulations codify NASA's legal commitment to integrate the NEPA process into program and project formulation.

NEPA is a procedural process that considers the potential adverse effects that proposed actions could have on human health and the environment. According to CEQ, actions include new and continuing activities, including projects and programs *entirely or partly financed, assisted, conducted, regulated, or approved* by Federal agencies. CEQ has established three levels of NEPA analysis based on the "context and intensity" of the potential adverse effects:

- Categorical l Exclusions (CATEXs) apply to actions not expected to individually or cumulatively have an adverse effect on human health and the environment. In 2012, NASA expanded the Agency NEPA regulations with 23 CATEXs grouped into five activities: administrative, operations and management, research and development, personal and real property, and aircraft and airfield. The majority of NASA's actions fall within a CATEX. A Record of Environmental Consideration (REC) is typically used to document a CATEX. Since the effects of these types of actions have already been evaluated by CEQ, RECs are not circulated for public review.
- Environmental Assessment (EA) documents analyze whether a proposed action could have a significant impact on the environment. If the analysis identifies no significant impacts, the decision is documented as a Finding of No Significant Impact (FONSI) signed by the Center Director or Associate Administrator for Headquarter actions. If potential significant impacts are identified that cannot be avoided or mitigated, an Environmental Impact Statement (EIS) needs to be prepared. NEPA requires public review of the draft EA and is required to consider comments received from regulators and the public.
- Environmental Impact Statements (EISs) are prepared for actions expected to have a significant impact on the environment. NEPA requires public scoping and is required to consider comments received on the draft EIS. The EIS includes the environmental analysis of the proposed action and any reasonable alternatives that have been identified. The EIS concludes with a Record of Decision (ROD) signed by the responsible Associate Administrator at Headquarters. Note: NASA is not obligated to select the alternative with the least adverse impact on the environment. NEPA only obligates the Agency to document that it has considered alternatives and their impacts before deciding to implement the action.

- EO 12114 is not mandated by NEPA, but needs to be considered whenever NASA contemplates an action with the potential for adverse effects outside the territorial jurisdiction of the United States. If the environmental evaluation indicates such effects are not significant under NEPA regulations
  (40 C.F.R. subpart 1508.27), the Center NEPA Manager (CNM) will assist the program or project manager in preparing a Memorandum For Record (MFR). The NASA Office of International and Interagency Relations (OIIR) will assist if international notice or outside concurrence is required.

NASA has issued NEPA policy in NPR 8580.1, NASA National Environmental Policy Act Management Requirements. The NPR specifies who has NEPA responsibilities and outlines what they are. To facilitate NEPA compliance, NASA has designated the CNM to assist in completing the Center environmental checklist to determine if a CATEX can be applied. Even if a program or project manager is certain their action will have no adverse effect, it is important for them to contact the CNM.

The Center Environmental Management Office (EMO) is responsible for documenting and tracking environmental requirements such as permits, chemical inventory, hazardous waste management and disposal, regulatory tracking, and reporting. The CNM uses the environmental checklist to confirm that the action falls within existing Center permits and conditions and to track the cumulative impacts of all Center actions. The checklist is also used to ensure that the action does not trigger an extraordinary circumstance listed in NASA NEPA regulations. The checklist applies to Center actions whether they are conducted on or off Center property. The NASA NEPA manager at Headquarters is the point of contact for actions not involving a Center, such as funding or grants directed to industry or a university.

The CNMs and NASA NEPA manager are tasked with supporting missions by expediting the NEPA process. For Radioisotope Power System (RPS)-enabled missions, the NEPA process is coordinated with the Nuclear Launch Safety Approval (NLSA) process requirements summarized in Section 7.4.2.

The Agency's NEPA program is managed by the NASA NEPA manager within the Office of Strategic Infrastructure, Environmental Management Division (EMD), HQ. The NEPA program maintains a NEPA desk guide and an internal repository of NEPA documents found in the EMD's NASA Environmental Tracking System (NETS) NEPA module. This repository updates the public NEPA library on the Agency website with final EAs (3-6 completed per year) and EISs (one completed every year or two). The website (www.nasa.gov/agency/nepa/) provides the NEPA library and contact information for the CNMs and NASA NEPA manager.

7.4.2 Nuclear Launch Safety Approval

Nuclear Launch Safety Approval (NLSA) is required for the launch of any quantity or type of radioactive material. This approval process applies to any mission that carries radioactive materials on the spacecraft, including but not limited to those used for calibration, power generation, or thermal management. The process involves an assessment of mission radiological risk that provides the basis for a decision whether to authorize the launch of the radioactive materials. Approval authority is delegated at several levels and is dependent upon the type and

quantity of radioactive material involved. Approval authority may extend as far as the Executive Office of the President. The type and quantity of radioactive material proposed for flight also determines the scope and depth of the analyses required for review and approval. Review requirements can range from simple notification to the NASA Office of Safety and Mission Assurance of the radioactive material to be launched, to an interagency review process involving NASA, the Department of Defense (DOD), Department of Energy (DOE), Environmental Protection Agency (EPA), and Nuclear Regulatory Commission (NRC).

Specific details concerning these requirements can be found in NPR 8715.3, NASA General Safety Program Requirements.

For any U.S. space mission involving the use of radioisotope power systems, radioisotope heater units, or nuclear reactors, launch approval must be obtained from the Office of the President per Presidential Directive/National Security Council Memorandum No. 25 (PD/NSC-25), "Scientific or Technological Experiments with Possible Large-Scale Adverse Environmental Effects and Launch of Nuclear Systems into Space," paragraph 9, as amended May 8, 1996. The approval decision is based on an established and demonstrated review process that includes an independent evaluation by an *ad hoc* Interagency Nuclear Safety Review Panel (INSRP) comprised of representatives from NASA, DOE, DOD, and EPA, with an additional technical advisor from the NRC. The process begins with development of a launch vehicle databook (i.e., a compendium of information describing the mission, launch system, and potential accident scenarios, including their resulting environments and probabilities). DOE uses the databook to prepare a Preliminary Safety Analysis Report (PSAR) for the space mission. In all, three Safety Analysis Reports (SARs) are typically produced and submitted to the mission's INSRP: the PSAR, a draft final SAR (draft FSAR), and a final SAR (FSAR). The DOE project office responsible for providing the nuclear power system develops these documents.

The *ad hoc* mission INSRP conducts its nuclear safety/risk evaluation and documents its results in a Safety Evaluation Report (SER). The SER contains an independent evaluation of the mission's radiological risk. DOE uses the SER as its basis for accepting the SAR. If the DOE Secretary formally accepts the SAR-SER package, it is forwarded to the NASA Administrator for use in the launch approval process.

NASA distributes the SAR and SER to the other cognizant Government agencies involved in the INSRP, and solicits their assessment of the documents. After receiving responses from these agencies, NASA conducts internal management reviews to address the SAR and SER, the external assessments of them, and any other nuclear safety information pertinent to the launch. If the NASA Administrator decides to proceed with the nuclear safety launch approval process, the NASA Administrator sends a request for nuclear safety launch approval to the director of the Office of Science and Technology Policy (OSTP) within the Executive Office of the President.

NASA HQ is responsible for implementing this process for NASA missions. It has traditionally enlisted the Jet Propulsion Laboratory (JPL) to assist in this activity. DOE supports the process by analyzing the response of power system hardware to the different accident scenarios identified in the databook, and by preparing a probabilistic risk assessment of the potential radiological consequences and risks to the public and the environment for mission accident scenarios. NASA's Kennedy Space Center (KSC) is responsible for overseeing development of

databooks, and traditionally uses JPL to characterize accident environments and integrate the databooks. KSC and JPL subcontractors both provide information relevant to supporting the databook development. The development team ultimately selected for a mission would be responsible for providing payload descriptions, describing how the nuclear hardware integrates into the spacecraft, describing the mission and reasonable alternatives, and supporting KSC and JPL in their development of databooks.

NASA Mission Directorate Associate Administrators (MDAAs), Center Directors, and program executives involved with the control and processing of radioactive materials for launch into space should ensure that the basic designs of vehicles, spacecraft, and systems utilizing radioactive materials provide protection to the public, the environment, and users, such that radiation risk resulting from exposures to radioactive sources are as low as reasonably achievable. Nuclear safety considerations should be incorporated from the Pre-Phase A concept study stage throughout all project stages to ensure that the overall mission radiological risk is acceptable. All space flight equipment (including medical and other experimental devices) that contain or use radioactive materials should be identified and analyzed for radiological risk. Site-specific ground operations and radiological contingency plans should be developed commensurate with the risk represented by the proposed launch of nuclear materials. Contingency planning, as required by the National Response Framework, includes provisions for emergency response and support for source recovery efforts. Specific details concerning these requirements can be found in NPR 8710.1, Emergency Preparedness Program, and NPR 8715.2, NASA Emergency Preparedness Plan Procedural Requirements.

7.4.3 Risk Communication

HQ/EMD objectives include advancing NASA's environmental stewardship, identifying and mitigating potential environmental consequences of project activities, complying with existing environmental regulations and statutes, and performing any required environmental clean-up that results from historic practices. In all of these areas, open and ongoing public communications are critical to establishing stakeholder awareness, understanding, and endorsement of NASA's activities.

The risk communication process encompasses the development, review, and dissemination of information products and trained spokespeople capable of addressing the aspects of NASA missions that have the potential for generating environmental or safety concerns among the general public, media, or Government.

Past NASA experience has shown that the NEPA and Nuclear Launch Safety Approval processes can be significantly more effective when they include early and continuous consideration of risk communication principles, policies, and procedures, from project start to completion. Those principles are strongly anchored in the recognition of the public's role in weighing the value and safety of major federally funded national endeavors. Based fundamentally on a two-way exchange of information between stakeholders, risk communication principles provide the following general guidance:

- **Be open**: maintain transparent decision-making processes

- **Be accurate**: ensure that project information is technically correct and dispensed by well-informed spokespersons
- **Be clear**: craft and widely disseminate easily understood information to explain the "whys" and "hows" of potentially hazardous or controversial actions
- **Be respectful**: be aware of different cultural perceptions and concerns, and
- **Be interactive**: invite public discussion early and often using a wide variety of channels, and expect to receive inquiries.

A formal risk communication strategy (coupled to the project's community outreach/involvement effort) provides NASA with the best chance to successfully implement environmental stewardship programs and restoration and cleanup projects with the support of the local community and various stakeholder groups. Early involvement and communication increases collaboration and understanding, builds trust and credibility, and reduces the potential for conflict. Risk communication can also improve NASA's environmental risk management decision-making. A coordinated approach to risk communication facilitates more accurate and consistent information products across all audiences, more complete review and concurrence by all program partners, better preparation for key mission events, and an overall greater chance for mission success.

This approach also lowers costs for each individual program/project through extensive cost sharing, and helps to reduce unintended conflicts in information produced for and by individual projects. It enables NASA to provide timely, clear, and concise information on its projects and plans, fostering a more knowledgeable and involved stakeholder community. A risk communication plan is required for any program or project that involves the subject areas in Section 7.4.1. This plan typically includes a variety of internal and external information products (talking points, responses-to-queries, and frequently asked questions), detailed review processes and contacts for these products, identification and training of key project spokespeople, and—in the case of launches with radioactive materials—active preparation and operation of a joint information center to support radiological contingency planning.

Further detail regarding NEPA compliance requirements for NASA programs and projects can be found in NPR 8000.4, Agency Risk Management Procedural Requirements. NASA risk communication is governed by the policies and processes in the Risk Communication Plan for Planetary and Deep Space Missions of the NASA Science Mission Directorate (1999).

7.4.4 Planetary Protection

The United States is a signatory to the United Nations' Treaty of Principles Governing the Activities of States in the Exploration and Use of Outer Space, Including the Moon and Other Celestial Bodies. Known as the Outer Space Treaty, it states in part (Article IX) that exploration of the Moon and other celestial bodies shall be conducted "so as to avoid their harmful contamination and also adverse changes in the environment of the Earth resulting from the introduction of extraterrestrial matter." NASA policy (NPD 8020.7, Biological Contamination Control for Outbound and Inbound Planetary Spacecraft) specifies that the purpose of preserving solar system conditions is for future biological and organic constituent exploration. This NPD also establishes the basic NASA policy for the protection of the Earth and its biosphere from

planetary and other extraterrestrial sources of contamination. The general regulations to which NASA flight projects should adhere are set forth in NPR 8020.12, Planetary Protection Provisions for Robotic Extraterrestrial Missions. Different requirements apply to different missions, depending on which solar system object is targeted or encountered and the spacecraft or mission type (flyby, orbiter, lander, sample return, etc.). For some bodies (such as the Sun and Mercury), there are minimal planetary protection requirements. Current requirements for the outbound phase of missions to Mars and Europa, however, are particularly rigorous. Table 7.4-1 shows the current planetary protection categories, while Table 7.4-2 provides a brief summary of their associated requirements. Documentation for human exploration is being developed. The Committee on Space Research (COSPAR) has guidelines written in their policy and NASA has released NPI 8020.7 NASA Policy on Planetary Protection Requirements for Human Extraterrestrial Missions and a subsequent NPR for human exploration is planned.

At the core, planetary protection is a project management responsibility and a systems engineering activity. The effort cuts across multiple WBS elements. Failure to adopt a viable planetary protection approach and incorporate it into system engineering processes during the early planning phases will add cost, complexity, and potentially, schedule to the mission. Planning for planetary protection begins in Pre-Phase A, during which feasibility of the mission is established. Project managers should request a preliminary categorization letter in Pre-Phase A. There is a potential that a launch may be jeopardized if planetary protection requirements are not met. Prior to the end of Phase A, the project manager should send a letter to the Planetary Protection Officer (PPO) stating the mission type and planetary targets and requesting that the mission be assigned a planetary protection category.

Prior to the PDR, at the end of Phase B, the project manager should submit to the NASA PPO a planetary protection plan detailing the actions that will be taken to meet the requirements. Depending on the mission category, additional subsidiary plans may be required such as a Contamination Analysis plan, a Microbiological Assay Plan, and a Microbial Reduction Plan (See NPR 8020.12 for a complete list of required plans and when they are due). If a mission needs to be extended, an Extended Mission plan/Request will be needed to make the decision. The project's progress and completion of the requirements are reported in a planetary protection pre-launch report submitted to the NASA PPO for approval. The approval of this report at the FRR constitutes the final planetary protection approval for the project and should be obtained for permission to launch. An update to this report, the planetary protection post-launch report, is prepared to report any deviations from the planned mission due to actual launch or early mission events. For sample return missions only, additional reports and reviews are required prior to launch toward the Earth, prior to commitment to Earth reentry, and prior to the release of any extraterrestrial sample to the scientific community for investigation. Finally, at the formally declared End of Mission (EOM), a planetary protection EOM report is prepared. This document reviews the entire history of the mission in comparison to the original planetary protection plan and documents the degree of compliance with NASA's planetary protection requirements. This document and periodic mission status is typically reported on by the NASA PPO at a meeting of the Committee on Space Research (COSPAR) to inform other spacefaring nations of NASA's degree of compliance with international planetary protection requirements.

For additional information on planetary protection including the requirements for reviews and how deviations are handled, see NPR 8020.12.

Table 7.4-1 Planetary Protection Mission Categories

Planet Priorities	Mission Type	Category	Example
Not of direct interest for understanding the process of chemical evolution. No protection of such planets is warranted (no requirements).	Any	I	Flyby, Orbiter, Lander): Undifferentiated, metamorphosed asteroids; Io
Of significant interest relative to the process of chemical evolution, but only a remote chance that contamination by spacecraft could jeopardize future exploration.	Any	II	Venus; Moon (with organic inventory); Comets; Asteroids; Jupiter; Jovian Satellites except Io, Ganymede and Europa; Saturn; Saturnian Satellites other than Titian and Enceladus; Uranus; Uranian Satellites; Neptune; Neptunian Satellites other than Triton; Pluto/Charon; Kuiper-Belt Objects (e.g., Stardust outbound, Genesis (outbound), Cassini)
Of significant interest relative to the process of chemical evolution and/or the origin of life and for which scientific opinion provides a significant chance that contamination by spacecraft could compromise future investigations.	Flyby, Orbiter	III	Orbiters of Mars; Europa; Enceladus (e.g., Odyssey, Mars Global Surveyor, Mars Reconnaissance)
	Lander, Probe	IV	Landers for Mars; Europa; Enceladus (e.g., Phoenix Europa Explorer Mars Sample Return (outbound))
Any solar system body.	Unrestricted Earth return[a]	V	Unrestricted Earth Return (e.g., Stardust (return), Genesis (return))
	Restricted Earth return[b]	V	Restricted Earth Return (e.g., Mars Sample Return (return))

a. No special precautions needed for returning material/samples back to Earth.
b. Special precautions need to be taken for returning material/samples back to Earth. See NPR 8020.12.

Table 7.4-2 Summarized Planetary Protection Requirements

Mission Category	Summarized Requirements
I	Certification of category.
II	Avoidance of accidental impact by spacecraft and launch vehicle. Documentation of final disposition of launched hardware.
III	Stringent limitations on the probability of impact. Requirements on orbital lifetime or requirements for microbial cleanliness of spacecraft.
IV	Stringent limitations on the probability of impact and/or the contamination of the object. Microbial cleanliness of landed hardware surfaces directly established by bioassays.

V	Outbound requirements per category of a lander mission to the target. Detailed restricted Earth return requirements will depend on many factors, but will likely include sterilization of any hardware that contacted the target planet before its return to Earth, and the containment of any returned sample.

7.5 Use of the Metric System

The decision whether a project or program could or should implement the System Internationale (SI), often called the "metric system," requires consideration of a number of factors, including cost, technical, risk, and other programmatic aspects.

The Metric Conversion Act of 1975 (Public Law 94-168) amended by the Omnibus Trade and Competitiveness Act of 1988 (Public Law 100-418) establishes a national goal of establishing the metric system as the preferred system of weights and measures for U.S. trade and commerce. NASA has developed NPD 8010.2, Use of the SI (Metric) System of Measurement in NASA Programs, which implements SI and provides specific requirements and responsibilities for NASA.

However, a second factor to consider is that there are possible exceptions to the required implementation approach. Both EO 12770 and NPD 8010.2 allow exceptions and, because full SI implementation may be difficult, allow the use of "hybrid" systems. Consideration of the following factors will have a direct impact on the implementation approach and use of exceptions by the program or project.

Programs or projects should do analysis during the early life-cycle phases when the design solutions are being developed to identify where SI is feasible or recommended and where exceptions will be required. A major factor to consider is the capability to actually produce or provide metric-based hardware components. Results and recommendations from these analyses should be presented by SRR for approval.

In planning program or project implementation to produce metric-based systems, issues to be addressed should include the following:

- Interfaces with heritage components (e.g., valves, pyrotechnic devices, etc.) built to English-based units:
  - Whether conversion from English to SI and/or interface to English-based hardware is required.
  - The team should review design implementation to ensure there is no certification impact with heritage hardware or identify and plan for any necessary re-certification efforts.
- Dimensioning and tolerancing:
  - Can result in parts that do not fit.
  - Rounding errors have occurred when converting units from one unit system to the other.
  - The team may require specific additional procedures, steps, and drawing Quality Assurance (QA) personnel when converting units.
- Tooling:

- Not all shops have full metric tooling (e.g., drill bits, taps, end mills, reamers, etc.).
- The team needs to inform potential contractors of intent to use SI and obtain feedback as to potential impacts.

- Fasteners and miscellaneous parts:
  - High-strength fastener choices and availability are more limited in metric sizes.
  - Bearings, pins, rod ends, bushings, etc., are readily available in English with minimal lead times.
  - The team needs to ascertain availability of acceptable SI-based fasteners in the timeframe needed.
- Reference material:
  - Some key aerospace reference materials are built only in English units, e.g., MIL-HDBK-5 (metallic material properties), and values will need to be converted when used.
  - Other key reference materials or commercial databases are built only in SI units.
  - The team needs to review the reference material to be used and ensure acceptable conversion controls are in place, if necessary.
- Corporate knowledge:
  - Many engineers presently think in English units, i.e., can relate to pressure in Pounds per Square Inch (PSI), can relate to material strength in Kilopounds per Square Inch (KSI), can relate to a tolerance of 0.003 inches, etc.
  - However, virtually all engineers coming out of school in this day and era presently think in SI units and have difficulty relating to English-based units such as slugs (for mass) and would require retraining with attendant increase in conversion errors.
  - The team needs to be aware of their program-or project-specific knowledge in English and SI units and obtain necessary training and experience.
- Industry practices:
  - Certain industries work exclusively in English units, and sometimes have their own jargon associated with English material properties. The parachute industry falls in this category, e.g., "600-lb braided Kevlar line."
  - Other industries, especially international suppliers, may work exclusively in metric units, e.g., "30-mm-thick raw bar stock."
  - The team needs to be aware of these unique cases and ensure both procurement and technical design and integration have the appropriate controls to avoid errors.
- Program or project controls: The team needs to consider, early in the SE process, what program- or project-specific risk management controls (such as configuration management steps) are required. This will include such straightforward concerns as the conversion(s) between system elements that are in English units and those in SI units or other, more complex issues.

Several NASA projects have taken the approach of using both systems, which is allowed by NPD 8010.2. For example, the Mars soil drill project designed and developed their hardware using English-based components, while accomplishing their analyses using SI-based units. Other small-scale projects have successfully used a similar approach.

For larger or more dispersed projects or programs, a more systematic and complete risk management approach may be needed to successfully implement an SI-based system. Such things as standard conversion factors (e.g., from pounds to kilograms) should be documented, as should standard SI nomenclature. Many of these risk management aspects can be found in such documents as the National Institute of Standards and Technology's *Guide for the Use of the International System of Units (SI)* and the DOD *Guide for Identification and Development of Metric Standards*.

Until the Federal Government and the aerospace industrial base are fully converted to an SI-based unit system, the various NASA programs and projects will have to address their own level of SI implementation on a case-by-case basis. It is the responsibility of each NASA program and project management team, however, to comply with all laws and executive orders while still maintaining a reasonable level of risk for cost, schedule, and performance.

7.6 Systems Engineering on Multi-Level/Multi-Phase Programs

Most of the examples in the preceding sections pertain to the basic Systems Engineering (SE) approach used for a single project as might be represented by single spacecraft managed by a single NASA Center and deployed into orbit by a single launch vehicle. That type of SE approach is referred to in this section as a Single Level, Single Phase (SL/SP) approach since the majority of NASA SE work is performed by a single level of management (i.e., a project office at a single NASA Center) and since the mission is deployed for operation in a single temporal phase (i.e., by a single launch). The following discussion illustrates how that SL/SP approach might be adapted when conducting a program wherein the majority of NASA SE work for the system of interest is performed by multiple levels of NASA management (i.e., a NASA Headquarters program office supported by project offices at multiple NASA Centers) and wherein the system of interest is deployed for operation in multiple discrete stages over time. The latter is referred to below as a Multi-Level, Multi-Phase (ML/MP) approach.

The multidimensional nature of SE on an ML/MP program is illustrated in this section through the use of a notional reference model that highlights unique considerations in multi-level SE management and in multi-phase design and assembly. Also addressed are related topics in ML/MP development including the heightened importance of concept design, experience desired for Systems Engineering and Integration (SE&I) team leaders, and commercial analogs.

7.6.1 Notional Reference Model

The notional reference model includes both an example system of interest and an example SE management approach. The system of interest (referred to below as the "system") is a notional space station, and the SE management approach is one in which NASA has the lead for Systems Engineering and Integration (SE&I), and there is no program-level prime contractor.

7.6.2 Management Hierarchy Nomenclature

Figure 7.6-1 shows the management levels typically used for ML/MP programs. Levels I and II represent NASA Headquarters program management, as performed by the program executive/program director and program manager, respectively. The Level II program manager and program office may be physically located at a NASA Center (i.e., the "lead Center" for the program) or other facility not co-located with Level I. Level III represents NASA project management as performed by project managers at project offices located at NASA Centers. While the Level I, II, and III nomenclature has some application to SL/SP projects as well, the distinction is that ML/MP programs are typically tightly coupled programs (see NPR 7123.1, fig. 5-2) wherein Level II is involved in day-to-day program management activities such as those discussed in Section 7.6.4**.**

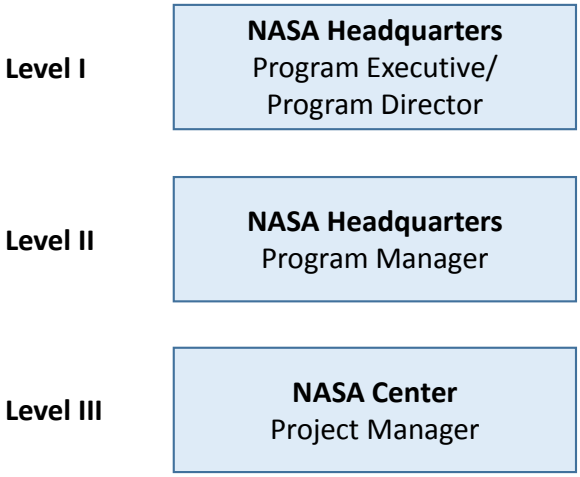


Figure 7.6-1 Management Level Hierarchy for ML/MP Programs

7.6.3 Multi-Dimensional Nature of SE

Systems engineering teams on ML/MP programs employ the same basic SE techniques as used on SL/SP projects, but they employ them at multiple levels and over multiple phases in time (i.e., in multiple dimensions) in what is effectively an advanced form of basic SE. While it is important to conduct basic SE functions well on SL/SP projects, it becomes even more important to conduct them well on ML/MP programs as ML/MP programs typically have greater scope and complexity. Relative to SL/SP projects, ML/MP programs typically have: a) more interfaces and system configurations, b) more projects, partners, stakeholders, contracts, and agreements, and c) higher cost and higher political visibility.

This multi-dimensional nature influences many aspects of how SE is conducted and managed on ML/MP programs. A selection of these aspects is discussed below. Some of these aspects would apply to any system for which significant SE tasks are performed at both Level II and Level III or to any remote system or facility that is developed and deployed over a series of incrementally operating interim stages.

7.6.4 Multi-Level SE Management Considerations

7.6.4.1 NASA SE Roles at Levels I, II, and III

Figure 7.6-2 shows a notional organizational structure for an ML/MP program wherein NASA is acting as the program-level integrator, and wherein there is no program-level prime contractor. It builds upon the three levels of NASA management depicted in Figure 7.6-1. Program work is distributed in discrete Work Packages (WPs) to project offices, each at a different NASA Center. International Partners (IPs) and Commercial Partners (CPs) may also be present in ML/MP programs. While not considered Level III, IPs/CPs interface technically with the program office in much the same way as does Level III, including through Level II interface control documentation. However, IP/CP participation also is governed by Agency-level agreements, some of which may be reflected in Level I requirements.

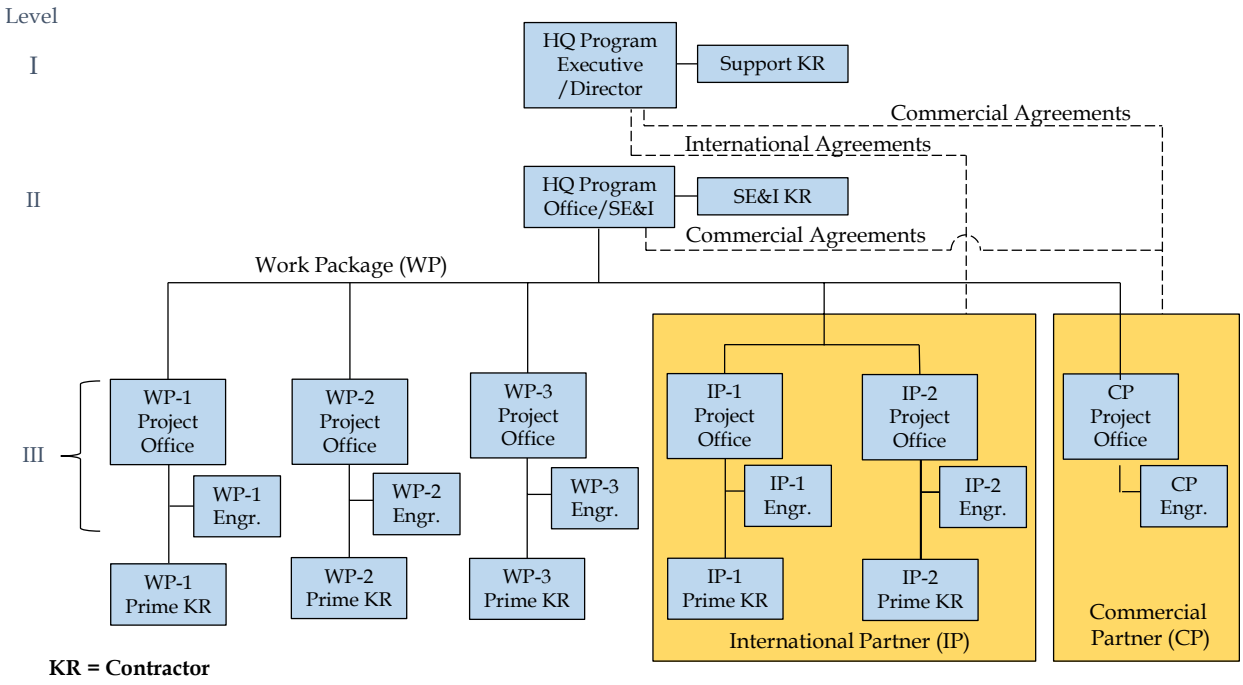


Figure 7.6-2 Notional Organization for an ML/MP Program wherein NASA is the Program-Level Integrator and wherein there is No Program-Level Prime Contractor

A summary of the NASA SE scope on ML/MP programs is given below. This summary excludes the significant SE work performed by WP prime contractors and subcontractors as they may be considered Level IV or lower-level team members.

7.6.4.1.1 Level I

NASA Level I typically controls key top-level program functional and performance requirements, key top-level schedule milestone dates, and top-level technical user resource allocations. Level I also controls international agreements, such as those for IP payload utilization. In addition, Level I may, along with Level II, control CP agreements. A support contractor staff may supplement the Level I office by participating in Systems Engineering and Integration (SE&I) activities and by conducting selected analyses for the program executive/program director.

7.6.4.1.2 Level II

The Level II program office (and its SE&I contractor) develops and delivers a verified and validated end-to-end system, by stage. (See Figure 7.6-3 for stage depictions.) The Level II program office develops and controls requirements (functional, performance, interface), technical resource allocations (e.g., mass, power, thermal, volume, extravehicular activity time, etc.), architectures, verification plans, development schedules, and operational sequences (including design reference missions) for the integrated end-to-end system by stage. It also conducts integrated functional and performance analyses (e.g., power, structural, thermal, attitude control, flight mechanics, utilization and operations assessments, etc.), life-cycle

reviews, and verification/validation for the integrated end-to-end system by stage. Due to their end-to-end scope, Level II analyses usually are broader and shallower than analyses conducted by Level III and serve a complementary role to Level III analyses. Level II life-cycle reviews are held to confirm integrated system design and operation meets Level II requirements, resource allocations, operational sequences, etc., when all Level III and IP/CP elements are integrated into stage configurations. Life-cycle reviews may address multiple stages when key capabilities are achieved. For example, if key capabilities are achieved at stages 3, 6, and 10, reviews might be held for stages 1-3, 4-6, and 7-10. The program office manages to technical, cost, and schedule requirements that flow down from key requirements and allocations managed at Level I. The Level II program office conducts a Level II Configuration Control Board (CCB), membership on which includes project managers from NASA Level III and IP project offices. Change requests typically may be initiated by NASA Level II, by the Level II SE&I contractor, by NASA Level III, or by an IP/CP. When initiated by NASA Level III or an IP/CP, the change first goes through the Level III or IP/CP CCB process, respectively.

7.6.4.1.3 Level III

Level III project offices develop and deliver verified and validated, launch-ready flight hardware and software end items, e.g., elements, utilities (utilities are discussed in Section 7.6.5), etc. Each project office develops and controls requirements (functional, performance, interface), designs, technical resource allocations (e.g., mass, power, thermal, volume, extravehicular activity time, etc.) verification plans, development schedules, and operational sequences for its end items and conducts end item-level functional and performance analyses, life-cycle reviews, and verification/validation at the element or utility level. Life-cycle reviews may address multiple stages, consistent with Level II reviews. Each project office manages to its own set of technical, cost, and schedule requirements that flow down from system-level requirements and allocations managed at Level II. Each project has contract deliverables and its own CCB. Center engineering organizations typically provide engineering support to work package project offices.

7.6.4.2 Program-Level SEMP

Having a comprehensive program-level (Level II) SEMP in place at the time the program's respective projects start Phase B (see Figure 3.0-4) is an essential enabler. The Level II SEMP establishes the who, what, where, when, and how of SE&I process implementation, a key step in delineating program-wide SE roles and responsibilities. Along with clearly identifying forums used for program-wide technical integration and technical decision-making, it facilitates important transparency in product development among projects as well as between projects and Level II SE&I. Baselined by program SDR (see Figure 3.0-1, 2 and 3), the Level II SEMP also serves as a guide for developing project-level SEMPs which are baselined by project SRR. While gaining agreement on program-wide SE roles and responsibilities can be challenging on multi-level programs, particularly when IPs and CPs are involved, deferring agreement on roles and responsibilities in an approved Level II SEMP can lead to even greater, recurring challenges. When multi-level programs are also multi-phase programs, these recurring challenges are likely to incur additional impacts as the effects of indecision on SE conduct for one stage are likely to reappear in multiple stages.

7.6.4.3 Technical Resource Allocation

One of the responsibilities of SE&I is to allocate technical resources (see Section 7.6.4.1) among projects to fit within available system capabilities. Resource allocations initially are established in Level II requirements at the end of concept design, consistent with Level I requirements. These allocations can be refined under Level II configuration control as the program evolves. Investing time in establishing a resource allocation approach that incentivizes project managers to negotiate with unneeded resources (e.g., WP-1 trades 100 kg of mass to WP-2 in exchange for 80 W of power for the mutual benefit of WP-1, WP-2, and the program) versus holding unneeded resources is recommended. Without such an approach, reallocating can be difficult and time consuming for programs with multiple projects, multiple stages, and international/CP agreements, even with best efforts among Level II and Level III SE teams. This is particularly true for programs that are also over-budget, behind schedule, and below advertised performance.

7.6.5 Multi-Phase Design and Assembly Considerations

Designing a system that grows by adding elements over time introduces unique SE considerations relative to those typical of more traditional SL/SP systems. A central characteristic of ML/MP development is that the system needs to be able to operate acceptably not only in the assembly complete configuration, but also in interim configurations during the assembly phase. Operation during interim configurations enables the system to survive and potentially to be productive (e.g., enables limited research) as it is being built.

Figure 7.6-3 illustrates such a scenario for a notional space station. Flight elements (FEs) are designed and tested on the ground, placed into a launch vehicle as launch packages (LPs), and assembled onorbit to existing, incrementally larger stages over time. The system is shown as it exists at stages 1 and 2 (interim configurations) and at stage 10 (assembly complete configuration).

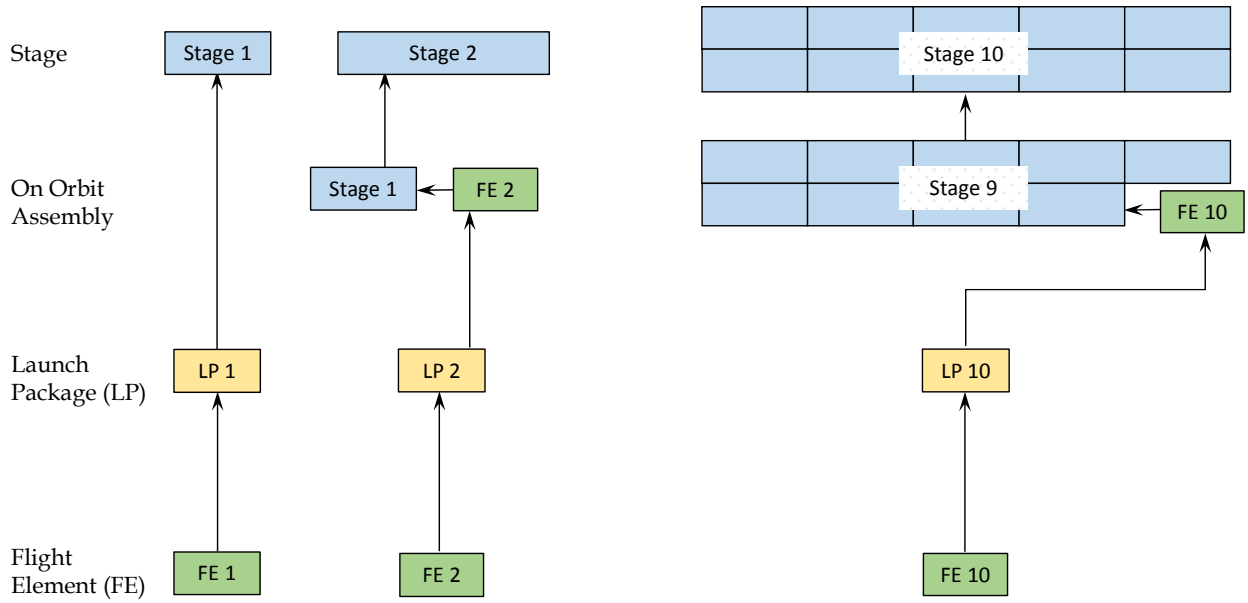


Figure 7.6-3 Flight Elements of a Notional Space Station being Designed, Launched and Assembled Onorbit into Incrementally Larger Stages over Time

Three key SE considerations in designing an ML/MP system (utility sizing, launch and assembly sequence, and on-orbit maintenance) are discussed below.

7.6.5.1 Utility Sizing

The need to function during multiple, incrementally growing stages of assembly can drive utilities on ML/MP systems to operate under an unusually wide range of conditions. Utilities might include power, thermal control, environmental control and life support, propulsion, attitude control, command and data handling, communication, etc. Utilities that provide bus-type services at interfaces to multiple elements across a space station-like facility typically are sized for the most demanding condition, usually that associated with the assembly-complete configuration with a full crew and with full mission and payload operations. However, performance needs to be verified not only for the most demanding condition, but also for a wide range of lower-load conditions at interim stages of assembly. For example, thermal trunk lines and pumps that transport thermal control system fluid from element (e.g., pressurized module) interfaces to radiators are sized to reject the required heat load at assembly complete, even though those trunk lines and pumps are operated under much lower-load conditions in prior stage configurations. Doing this avoids the need to retrofit trunk lines and pumps each time heat load from a new element is added. As a consequence, however, these trunk lines and pumps may be significantly oversized for operation at lower loads during early assembly stages.

Performance and operation of the utilities may also be affected by changes in external geometry as well as by varying attitudes required during assembly. For example, geometry changes and/or attitude changes can influence aspects such as power generation, thermal radiation, propulsion plume impingement, attitude control, flight mechanics, communication lines of sight, and approach corridors for launch vehicle rendezvous and docking.

7.6.5.2 Launch and Assembly Sequence

Choreographing the assembly sequence such that the required functional capability, fault tolerance, and maintainability is present at each stage while also effectively utilizing launch vehicle payload is an ever-present consideration in ML/MP systems. The system, e.g., flight elements, utilities, etc., needs to be incrementally manifested on the launch vehicle in a useful sequence to provide the required on-orbit capability during assembly. For example, if a habitable human presence is required at stage 3 to conduct selected experiments, stage 3 needs to have both a pressurized module and the required utilities. If that required human presence is permanent, the stage 3 utility capability needs to be sufficient to sustain the crew permanently while conducting selected experiments and to provide full fault tolerance for crew survival. If, however, that human presence is only temporary, i.e., the crew is aboard when a launch vehicle with full crew return capability is present, stage 3 may need neither the capability to support the crew permanently nor the capability to provide full fault tolerance for crew survival as the launch/return vehicle may be able to supplement the required stage 3 capability and fault tolerance.

Assembly sequence and SE complexities greatly increase with the number of stages. For example, at each stage, requirements that meet stakeholder needs have to be defined and controlled, the design has to be analyzed and verified, a concept of operations has to be defined to validate operations, etc. As the number of stages typically is driven by available launch capacity, the launch vehicle becomes a central consideration in ML/MP systems. Included in this consideration is launch vehicle reliability, noting that larger launch systems enable reduced on-orbit assembly time but also risk larger portions of system assets to a single launch failure.

7.6.5.3 On-Orbit Maintenance

As ML/MP systems are likely to be associated with programs that have relatively long on-orbit lives, they typically will need to be designed for on-orbit maintenance. Depending on the fault tolerance philosophy employed, components needed for time critical repair may need to be stored on board, whereas other components may be able to be provided by scheduled resupply flights. On-orbit maintenance during assembly also needs to be planned for, particularly for programs that have extended assembly phases.

7.6.6 Additional SE Considerations for ML/MP Programs

7.6.6.1 Heightened Importance of Concept Design

Due to the increased cost, complexity, and visibility associated with ML/MP programs discussed in Section 7.6.3, it is especially important that each project within an ML/MP program conduct a rigorous concept design study in Pre-Phase A and Phase A (see Figure 3.0-4) to establish a credible technical, cost, and schedule baseline prior to formal project start at the beginning of Phase B, and that the respective baselines among all projects integrate effectively to meet program-level requirements. Requirements, design, assembly sequence, etc., are relatively easy to change during pre-Phase A and Phase A wherein design teams are relatively small and organizationally flat. But once prime contracts, international and commercial agreements, and formal configuration control are in place in Phase B, requirements become significantly more challenging and costly to change. The more Centers, IPs, CPs, and stages are involved, and the later requirements are changed, the more challenging requirements changes become. In addition, requirements changes usually take longer to implement on ML/MP programs (relative to SL/SP projects) due to the multi-level CCB structure. Project-level changes initiated at Level III often need to go through both Level III and Level II CCBs, and program changes initiated at Level II typically need to be evaluated for potential impacts by Level III.

To illustrate complexities that may arise when an ML/MP program hasn't achieved a credible technical, cost, and schedule baseline at Phase B start for its constituent projects, consider a case wherein the notional space station experiences a significant cost overrun and needs to make a major design descope to stay within budget. Instead of building out to the original assembly-complete configuration at stage 10, the funded (descoped) baseline stops at a major interim capability at stage 6. The program retains informal plans for building out to the stage 10 configuration (now considered the "extended baseline") should additional funds become available. However, this presents a dilemma to the SE team. If the program designs meet requirements only up to the funded baseline (stage 6), the utilities may be undersized for the extended baseline (stage 10). Alternatively, if the program retains the extended baseline as its design point, the utilities may be unnecessarily oversized for the funded baseline configuration,

should additional funds not become available. At a minimum, this type of uncertainty in the design point can lead to time-consuming configuration control challenges and risks for SE&I teams on ML/MP programs. Given the large scale of these programs, it could also lead to significant issues if configuration control is lost.

7.6.6.2 Experience Desired for SE&I Team Leaders

As a large portion of the SE team may be working on its first ML/MP program and may be finding the environment relatively unfamiliar and uncertain, it is desirable that program level SE&I leadership on ML/MP programs have both a mastery of basic SE techniques from SL/SP projects (across all life-cycle phases) and significant experience in a previous ML/MP program. Having this skill set will help the SE&I team leadership proactively and effectively manage uncertainty and unfamiliarity while achieving program objectives. It will also help avoid on-the-job learning, which could result in potentially costly, program-wide impacts.

7.6.7 Commercial Analogs to ML/MP Development

While ML/MP programs such as the notional space station discussed above are infrequent in space systems, there may be more common (albeit partial) analogs among terrestrial commercial construction projects, particularly among those designed and built in discrete phases over time to enable increasing levels of operation during construction phases. For example, there may be a partial analog in the design, construction, and operation of a large facility that is segmented into discrete "wings," and which begins operations with the first wing while subsequent wings are either being designed or constructed. Systems engineering and integration expertise from such terrestrial projects may be of value to ML/MP program SE&I teams.

7.7 Fault Management

Fault Management (FM) is a consideration in NASA's systems engineering process in designing, developing and operating NASA missions that focuses on understanding and managing the off-nominal system behaviors. Fault management addresses the off-nominal (unintended and unexpected) behaviors of a system through the design of behaviors that protect the system functions. FM can be implemented in hardware, in software, or by operators of the flight system, ground system, or both. All operational systems undergo wear and many will experience faults due to design flaws and/or unanticipated conditions outside of the design. Consequences of degradations and faults can range from degraded efficiency to catastrophic failures. Timely and effective detection and mitigation of these conditions can make the difference between mission failure and success, expensive and cost-effective missions, and delayed and on-schedule missions. Therefore, the key for containing these detrimental effects is to include FM throughout the entire system life cycle.

FM has emerged and developed along several paths in response to NASA's mission needs (e.g., deep space missions, satellites, and human spaceflight) as reflected by the different FM approaches used across NASA and by the challenge to gain community consensus on the nomenclature (See *NASA-HDBK-1002, Fault Management Handbook*). However, all of these efforts have common goals, namely to build in system robustness and resiliency, preserve the system and reduce life-cycle costs, including avoidance of system failure and unavailability costs. Regardless of the FM approach, the designs are driven by system mission success criteria, safety, and resource constraints.

Effective FM implementation requires a system-level perspective as it is not merely a localized concern. FM is a crosscutting engineering discipline that requires an identified representation on the SE team and close coordination with other activities of the SE, SMA, and subsystem engineering teams. A system's design is not complete until potential failures are addressed. Comprehensive FM relies on the cooperative design and operation of separately deployed system elements (e.g., in the space systems domain, flight, ground, and operations deployments) to achieve overall reliability, availability, maintainability, and safety objectives. Like all other system elements, FM is constrained by programmatic and operational resources. Thus, FM practitioners are challenged to identify, evaluate, and balance risks to mission objectives against the cost of designing, developing, validating, deploying, and operating FM functionality.

This section provides a brief overview of FM capabilities, significance, and connections to the SE process. More detailed guidelines and recommendations for defining, developing, analyzing, evaluating, testing, and operating FM and suggested processes for developing FM throughout the life cycle of a mission for different project types can be found in NASA FM HDBK-1002, which provides a more detailed FM methodology.

7.7.1 Elements of Fault Management

FM encompasses functions that enable an operational system to prevent, detect, diagnose, identify, predict, and respond to anomalous and failed conditions interfering with intended operations. From a methodological perspective, FM includes processes to analyze, specify, design, verify, and validate these functions. From a technological perspective, FM consists of monitoring and control elements, often embodied in software and procedures, of an operational

system by which the FM capability is realized. This includes a situational awareness capability such as caution/warning functions to notify ground operators and flight crew members of anomalous conditions, potential or existing hazards, and automated responses. The goal of FM is the preservation of safety and mission success including system assets, crew, and intended system functionality (via design or active control) in the presence of predicted or existing failures and degradation.

Broadly speaking, FM consists of *monitoring*, *assessment*, and *fault mitigation and recovery* elements as shown in Figure 7.7-1. Guided by the FM strategy and system design, these elements are implemented through a variety of mechanisms (software, hardware and/or processes/procedures) on both flight and ground systems. For instance, a monitoring system may consist of sensors, transducers, gauges, probes, and data acquisition systems that facilitate further reasoning by the assessment module about system state. Given a state assessment, the course of further action is decided for fault mitigation and system recovery. Mitigation and recovery actions can be autonomous, automated, or human-controlled. Historically, Fault Detection, Isolation, and Recovery (FDIR) have been implemented on NASA missions. FDIR, as traditionally defined, represents a limited scope FM approach and needs to be understood more broadly. The functional elements of FM are briefly described next.

7.7.1.1 Monitoring

Monitoring is the first and most basic element of FM that allows collection of system data through installed sensing and communication infrastructure. It includes data logging, storage, and display functions. Monitored data are used in a variety of ways for inference about system states during or after operation through automated, manual, or mixed-mode processing. Monitored data storage is also often used for reconstructing scenarios for process improvement or post-operational investigations.

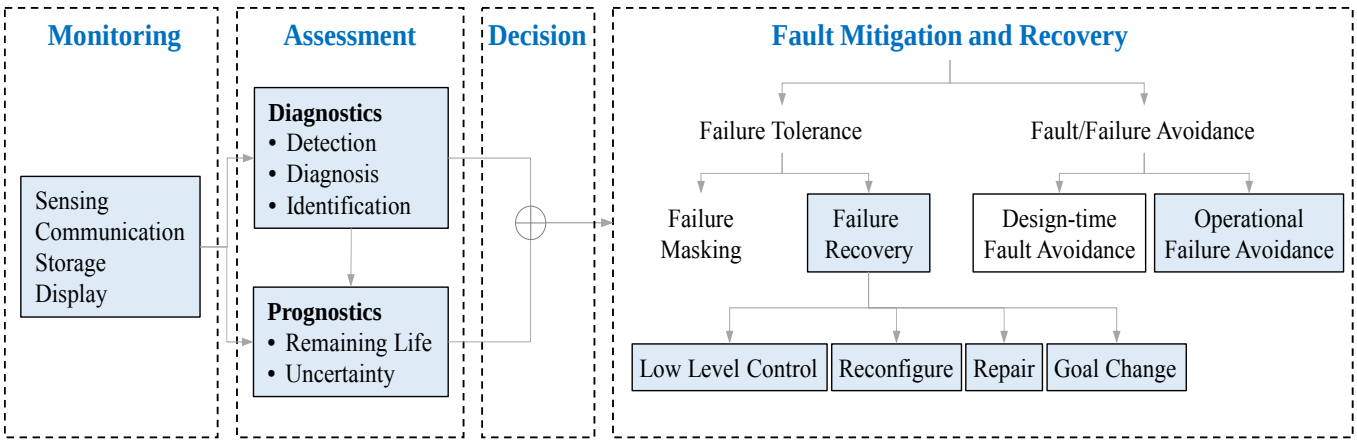


Figure 7.7-1 Functional Elements of a Fault Management System

7.7.1.2 Assessment

Assessment is the reasoning element that utilizes monitored data to determine the state of the system and generates information that guides decision-making for fault mitigation and recovery. Assessment includes determining system health states, determining causes of off-nominal

conditions (diagnostics), and predicting the evolution of a fault into the future (prognostics) leading to a potential failure.

7.7.1.2.1 Diagnostics

Diagnostics is a composite function that includes *detection*, *diagnosis,* and *identification* of a fault condition (*NASA- HDBK-1002, Fault Management Handbook*). Data are collected using on-board and remote sensors placed on the system "guided by FMECA[3] data, hazard analysis, service reports, and current design challenges" in order to provide accurate information on a system's state. More detail can be found in the *Integrated Vehicle Health Management* (IVHM): *Technology* book. (See Jennions 2013.) Potential faults are detected, diagnosed and identified using diagnostic algorithms either on board or remote. A minimal diagnostic system may contain early fault detection and initiate a safe mode to prevent further damage. Higher granularity diagnostics include automated fault diagnosis (pinpointing the root cause(s) of the fault or failure) and identification (determining the location of fault or failure).

7.7.1.2.2 Prognostics

Prognostics is the function used to predict or prognosticate a system's future condition, degradation, and determine Remaining Useful Life (RUL). This is done using data from sensors with results from onboard or remote diagnostics to assess system degradation over time. A prognostic model allows the quantification of time to failure, conditional on future operational activities (load) and environmental conditions. This information can then drive mitigation and other actions described under the fault mitigation and recovery element. The unique feature of prognostics is that it is likely to detect an impending failure before it actually happens, thereby allowing fault avoidance towards saving valuable time and reducing Loss Of Crew/Loss Of Mission (LOC/LOM) risks.

7.7.1.2.3 Decision Functions

The decision functions determine the required response to the fault assessment, selecting an action to mitigate current or future failure or effect thereof. These functions are implemented during the design phase to mitigate potential failures (e.g. through redundant units with active control or design margins). Functions implemented in the design phase map known potential failures to responses and determine the priority of suggested actions. Response decisions can also be implemented during operations in unanticipated situations through human intervention. Therefore, decision functions can be autonomous, human interactive, or a combination of automation with human intervention.

7.7.1.3 Fault Mitigation and Recovery

Mitigation and recovery generate recommendations and/or executing actions to mitigate harmful effects of faults and external hazards. The goal is to optimize overall system performance based on information from fault assessment functions, such as diagnostics and prognostics, and with a contextual understanding of prioritized mission goals.

---

[3] Failure Modes, Effects, and Criticality Analysis

Fault mitigation consists of prevention and tolerance, which can be implemented during design time or operations and can be in the form of autonomous, automated, or human controlled. As shown in Figure 7.7-1, mitigations are implemented using one or more of the following strategies:

- Design-Time Fault Avoidance: The function and FM capabilities are designed to minimize the risk of a fault and resulting failure using, for example, stricter quality assurance processes, higher quality parts, or increased margin.
- Operational Failure Avoidance: If a failure can be predicted, then action can be taken to prevent it from happening, generally through repair, replacement, or operational changes that reduce the failure's probability or delay its occurrence.

In fault tolerance, faults and failures are allowed to occur or sometimes are not avoidable, but their effects are mitigated or accepted in various ways that maximize mission safety and success:

- Failure Masking: Sometimes a lower-level failure can be allowed to occur when its effects can be masked so that it does not affect the higher-level system function.
- Failure Recovery: Sometimes a failure can be allowed to temporarily compromise the system function when it is possible to respond and recover before the failure compromises a mission goal. This may be implemented in various ways depending on decision time-scales and the complexity of the problem. For very short time-scales and lower complexity, low-level autonomous control strategies are adopted, whereas for larger time-scales and more complex situations, decisions are made with humans-in-the-loop. Some common recovery strategies are as follows:
  - Low-level control: Allow system controls to autonomously adjust system parameters to mitigate the effects of faults.
  - Reconfigure: Allow reconfiguration, taking advantage of physical or analytical redundancy in the system.
  - Repair: Allow system repairs wherever possible to mitigate faults.
  - Goal change: Allow a failure to compromise the system function, and respond by changing the system's goals to new, usually degraded goals that can be achieved.

It is important to note that although not all FM mitigations may be implemented for a given application, all FM functions are required to ensure system performance. Some FM systems may operate with a subset of FM mitigations, based on the mission needs. Limiting the set of mitigations could optimize the reduction of risk against life-cycle cost.

7.7.2 Fault Management and the Project Life-Cycle

FM addresses the off-nominal design and responses to failures and is developed in unison with the nominal system design, as shown in Figure 7.7-2. Mission and system characteristics, such as risk posture, response latency, fault tolerance requirements, and reliability requirements, drive the development process and the design. FM capabilities must be designed and implemented early in the design phases within and across subsystems, across hardware and software, and

across flight and ground systems. In contrast, on a number of past missions, the need to address fault management was not realized early enough in the life cycle, resulting in the system design crystallizing prior to identification of system deficiencies through analysis and testing. This resulted in a patchwork approach that generated higher-cost, ad hoc, suboptimal designs, gaps in coverage, difficulties in the V&V campaign, and overall increased risk. Incorporating FM early in the system design process enables a systematic, cost effective, and more capable FM approach. It also provides opportunities to influence the system design to design out potential vulnerabilities and avoid potential failures, to design monitors, and to identify the most cost-effective balance between mitigation, detection, and response. Therefore, as system complexity continues to grow and economic factors become increasingly important, the best approach is to consider the project's FM strategy from the conceptual design phase. The key activities parallel the SE process, including off-nominal scenario development, conceptual design, requirements development, architecture and design, assessment and analysis, V&V, and operations and maintenance, which are described in the following subsections.

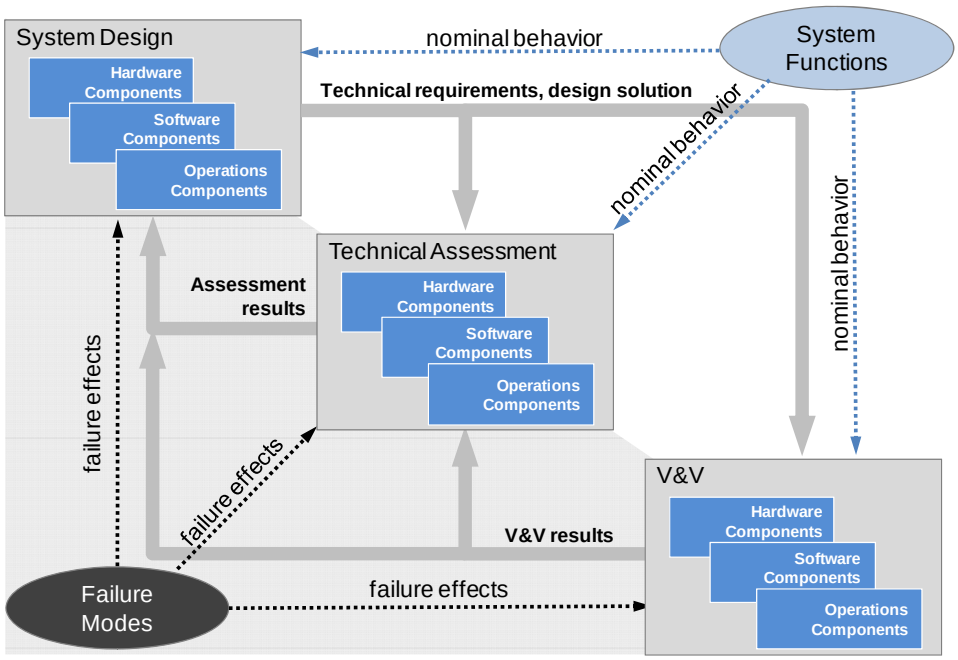


Figure 7.7-2 FM Follows a SE Process, Addressing Off-Nominal Conditions/Effects of Failures (Lower-Left) in Parallel with Activities to Achieve Nominal System Functions (Upper-Right)

7.7.2.1 Conceptual Design

The FM conceptual design activity includes defining the FM scope and philosophy and encompasses all elements of the system (i.e., hardware, software, and operations), all phases of the mission, all aspects of operating the system, the environment within which the system is required to operate, and the risk posture for the mission. This requires a thorough analysis to identify off-nominal scenarios that FM would target, and develop a corresponding concept of operation. Furthermore, design of appropriate user interfaces should be addressed that enables interaction with FM systems for autonomous control, manual control, or a combination as part of the conceptual design. The conceptual-design activity results in a baseline mission FM

architecture that meets the goals and objectives of the mission and is capable of being implemented within the resources allocated to the project.

For example, for a robotic Mars lander with low-risk tolerance and a critical sequence of events for EDL (Entry, Descent, and Landing), it may be possible to architect the system to manage faults with redundancy and fault masking due to the short time to criticality.

7.7.2.2 Requirements Development

FM requirements are captured during the FM requirements development activity, which presents a set of clear and concise mission-level engineering requirements allocated to systems (i.e., flight, ground, payload, and launch vehicle) and subsystems (i.e., hardware, software, mission operations, and crew), where appropriate. (See Figure 7.7-3.) FM requirements depend on the development of the mission technical concept, the FM concept, and the fault tolerance, safety, reliability, and availability requirements including requirements for test capabilities (e.g., fault injection in flight hardware and test benches) to ensure that test environments accommodate verification of individual FM software modules and failure scenario tests.

For example, defining the autonomous survivability early in the mission will have an impact on the system design in many critical subsystems. As shown in this example requirement: "The project shall be able to survive any single failure without any ground assistance for at least TBD for launch, TBD for cruise, TBD for other critical events." The time factor for surviving a fault during these events will impact flight hardware and software as well as ground operations procedures and responses.

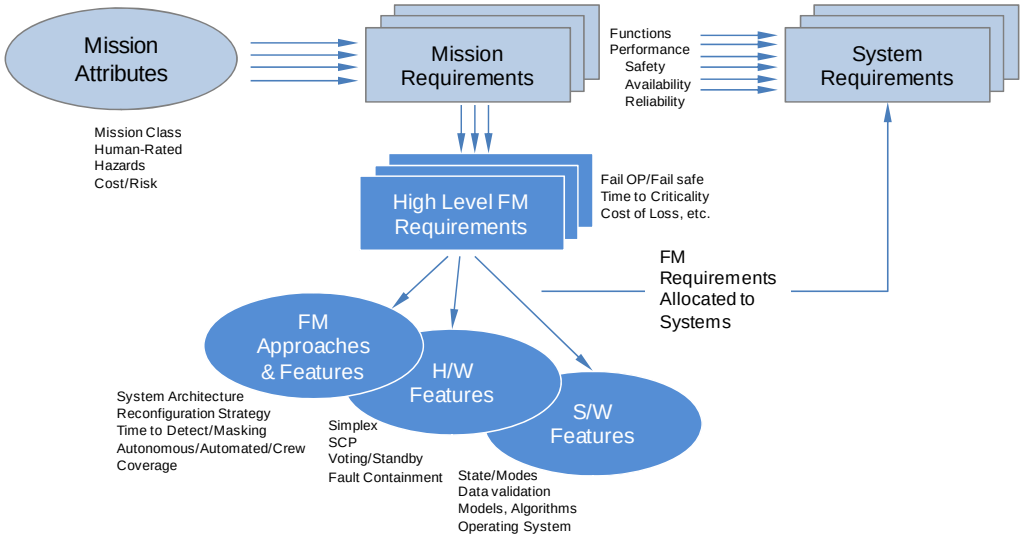


Figure 7.7-3 Deriving FM Requirements from Top-Level Mission Requirements and Allocating to Systems

7.7.2.3 Architecture and Design

The FM design process refines the FM requirements into a design that describes how failure conditions will be identified and what recovery steps will be taken. A technical specification called the FM Architecture Definition is developed, which defines how all allocated FM

responsibilities (defined under FM requirements) work together to address faults. Next, the focus of design activity is to identify potential adverse interactions, to define a system-level design that can implement the FM requirements, and to evaluate the adequacy of FM coverage.

By phase C of the project, the FM Design Specification is generated with detailed descriptions and diagrams of the failure monitors and responses, and includes the assumptions, failure potential, potential hidden states within each design description, monitor/response prioritization, and isolation and interaction prevention logic. Architecture and design documentation should include the following:

- Safing/abort design description;
- Failure detection, isolation, and recovery algorithms;
- Time-critical sequences design descriptions;
- ConOps for the use of redundancy;
- ConOps for pre-launch, ascent, post-launch; and
- ConOps for ground interaction, including diagnostics, repair, and recovery strategies.

7.7.2.4 Assessment and Analysis

This activity supports all phases of development by identifying possible faults/failures to be protected against and identifying possible response interactions or responses that may negatively impact another part of the system. It includes analyses to identify failures that can propagate outside a system boundary, prioritize limited resources (both processes and development), and devise mitigations to alleviate identified concerns.

7.7.2.5 Verification and Validation

An FM V&V plan, as a subset of the project V&V Plan, addresses the approach and risk posture to be taken for FM V&V. The plan documents guidelines, goals, and process steps for FM V&V actions that include test planning, plans for simulator development, test-bed certification, model accreditation, and identification of test assets and required fidelity. Validation and verification test matrices are generated as support documentation that serves as checklists to ensure that every requirement is included, and documents test procedures, test outcomes, and recommendations (for design changes, retest, or requirement waivers).

7.7.2.6 Operations and Maintenance

The operations plan is augmented with detailed FM-specific constraints and contingency procedures that implement the requirements allocated to ground and flight operations. Line-by-line procedures for interacting with the system during an unplanned or off-nominal event are included. For ground operations and flight crew operations, this includes maintenance and repair procedures, including diagnostics as applicable to the system. This planning activity addresses all mission phases, sequences, and modes when the FM system is used (e.g., pre-launch, launch, post-launch flight); FM transitions resulting from changes of phases, sequences, and modes; what needs to be done to perform check-out of the FM system; and plans for how to recover from safe modes or other off-nominal situations.

7.8 Technical Margins

7.8.1 Introduction

Implementors of successful projects have recognized the importance of establishing and managing technical margins to reduce development risk and increase the chance for mission success. Proper margin management is used to guide and govern system development and operations in order to mitigate the potential impacts of planned growth and unplanned growth. Margins are assigned to project specific technical metrics. As described in Section 6.7.2.6.2, technical metrics derive from Measures Of Performance (MOP) and Measures Of Effectiveness (MOE) and provide technical and programmatic leadership with necessary information to make informed decisions. While some technical metrics are outputs of program-controlled Models and Simulations (M&S), others are treated as aggregated margins across all subsystems and include planned (expected) growth and unplanned growth.

The main objective of this section is to provide definition, description, and guidelines for the identification and management of resource margins for space-flight projects, but the principles may be applied to airborne and ground system development projects as well. Many, if not all, NASA Centers have guidelines or requirements that address margin and, in some cases, growth allowance throughout the project life cycle. This section attempts to envelope the stated margins in those documents. In addition, the American Institute of Aeronautics and Astronautics (AIAA) has also developed standards for mass and power margins as documented in AIAA S-120-2006 and AIAA S–122-2007, respectively. These documents provide excellent guidance on the control of mass and power margins for space systems.

7.8.2 Definitions

Table 7.8-1 below provides definitions for commonly used terms, while Figure 7.8-1 provides a pictorial representation of the relationship of these terms to each other. The definitions are primarily derived from *AIAA S-120-2006, Mass Properties Control for Space Systems*. Although AIAA S-120-2006 is specific to the control of mass, the same terminology can be applied to most, if not all, other system characteristics or resources.

It is important for each project to define the terms it will use and use them in a consistent manner. This section does not imply that all the terms defined below need to be used, or that they have to be strictly adhered to, but they are provided as guidance.

Note that the use of *growth allowance* or *contingency* in addition to *margin* is often used only for mass. Typically, for all other resources only *margin* is used to account for both unplanned and planned growth of a resource.

Table 7.8-1 Definitions

Term	Definition
Basic Value	Basic Value, also known as Current Best Estimate (CBE) is based on the most recent baseline design or design concepts. For mass, this is the bottoms-up estimate of component mass as determined by the subsystem leads and is typically captured and tracked in the Master Equipment List (MEL). It includes an assessment of not yet defined design details.
Contingency	See definition for Growth Allowance below.
Current Best Estimate	Current Best Estimate (CBE), also known as the Basic Value for a given resource. See definition for Basic Value above.
Growth Allowance	Growth Allowance, also known as Contingency, accounts for the *expected* growth of a resource. The predicted change applied to the Basic Value of a resource (mass, power, etc.) based on an assessment of the design maturity, fabrication status of the item, and an estimate of the in-scope design changes that may still occur. Guidelines for Growth Allowance percentage are defined for each project phase, and are typically established in Phase A. % Growth Allowance = (Predicted Value – Basic Value) / Basic Value
Management Reserve	The resource budget reserved by management for out-of-scope and unplanned changes. Inside the project, it is typically controlled by the Program/Project Manager (PM), Lead or Mission Systems Engineer (LSE or MSE), or the program or project's Chief Engineer (CE). Outside the project, Management Reserve is typically controlled by NASA HQ, the program office, the launch vehicle provider, or the launch integrator.
Margin	Margin accounts for the *unexpected* growth in a resource over the project life cycle. It is the difference between the Resource Requirement and the Predicted Value. It is often referred to as a percentage of the Basic Value, Predicted Value, or Resource Requirement. % Margin = (Required Value – Predicted Value) / Basic Value (if growth allowance is not included in approach to margin management) % Margin = (Required Value – Predicted Value) / Predicted Value % Margin = (Required Value – Predicted Value) / Required Value
Predicted Value	The sum of the Basic Value and Growth Allowance. The Predicted Value is an estimate of the final value based on the current requirements and design.
Resource Limit	The maximum (or minimum) value of a resource (mass, power, etc.)that is imposed on a design due to contractual, performance, control, transport, or other requirements.
Resource Requirement	The limits against which Margins are calculated after accounting for Basic Value, Growth Allowance, and other uncertainties. Note: Derived from the requirements early in the design, the Resource Requirement (or Allowable Value) is intended to remain constant until there is a change to the requirements.

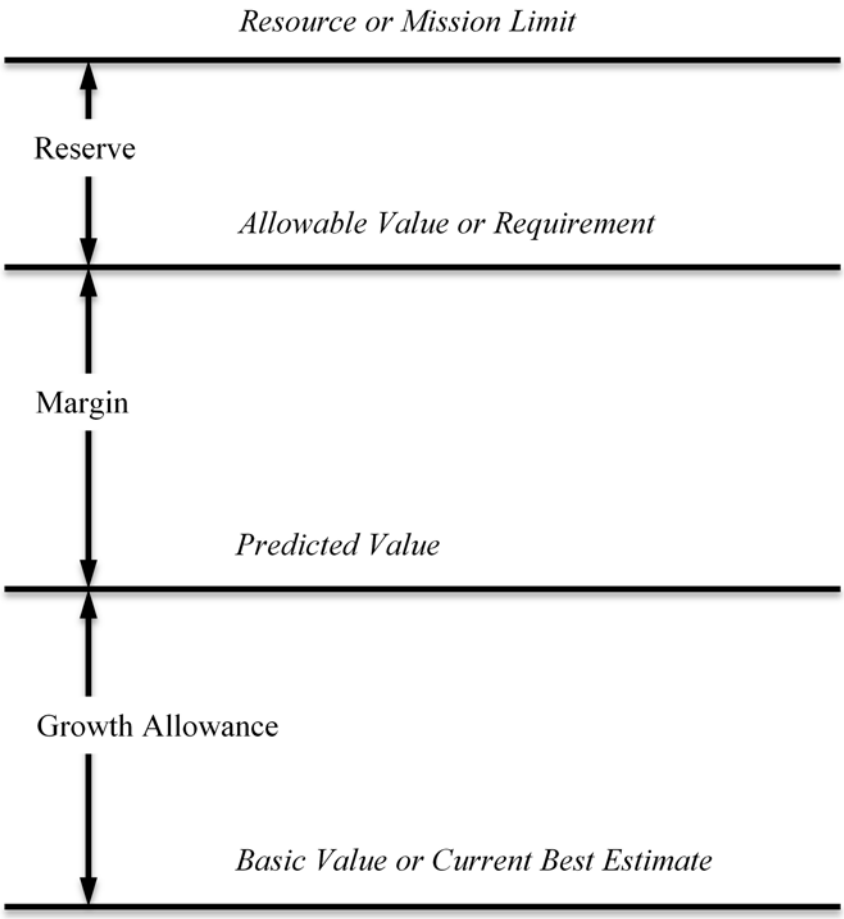


Figure 7.8-1 Definitions

7.8.3 Guidelines throughout the Project Life Cycle

7.8.3.1 Mass Margin and Mass Growth Allowance

Mass margin and mass growth allowance are closely related terms, but they account for different aspects of mass growth in the system. *Margin* accounts for unexpected growth while *Mass Growth Allowance* (MGA) accounts for expected growth. The early establishment of proper margins and growth allowances and the effective management of them throughout the project's life cycle play a critical role in the overall success of the mission.

The project should identify an allowance for the expected mass growth resulting from the lack of maturity in the current design. Mass growth typically varies as a function of hardware type and its development maturity. Development maturity can be thought of by project phase for newly developed or adapted components or by Technology Readiness Level (TRL). Mass growth allowance, sometimes known as contingency, should be applied at the lowest level tracked in the design or reported in the mass properties tracking system. Depletion of the growth allowance follows the phased design process; as the design and analyses of the hardware mature, the growth allowance depletes to reflect increased confidence in the predicted final mass. *AIAA S-120-2006,*

Mass Properties Control for Space Systems, provides guidance for growth allowance by development phase and subsystem type. In the absence of the development of project-specific growth allowances by subsystem, AIAA S-120-2006 provides good guidance or a point of departure for developing one.

The MGA percentages shown in AIAA S-120-2006 are applied to the basic value of each subsystem or component before margin is calculated. Margin calculated at the system level is directly dependent on the aggregate of subsystem basic mass plus subsystem or component MGAs, and should be applied according to the release timeline defined by the project over the project's life-cycle phases as depicted in Figure 7.8-2.

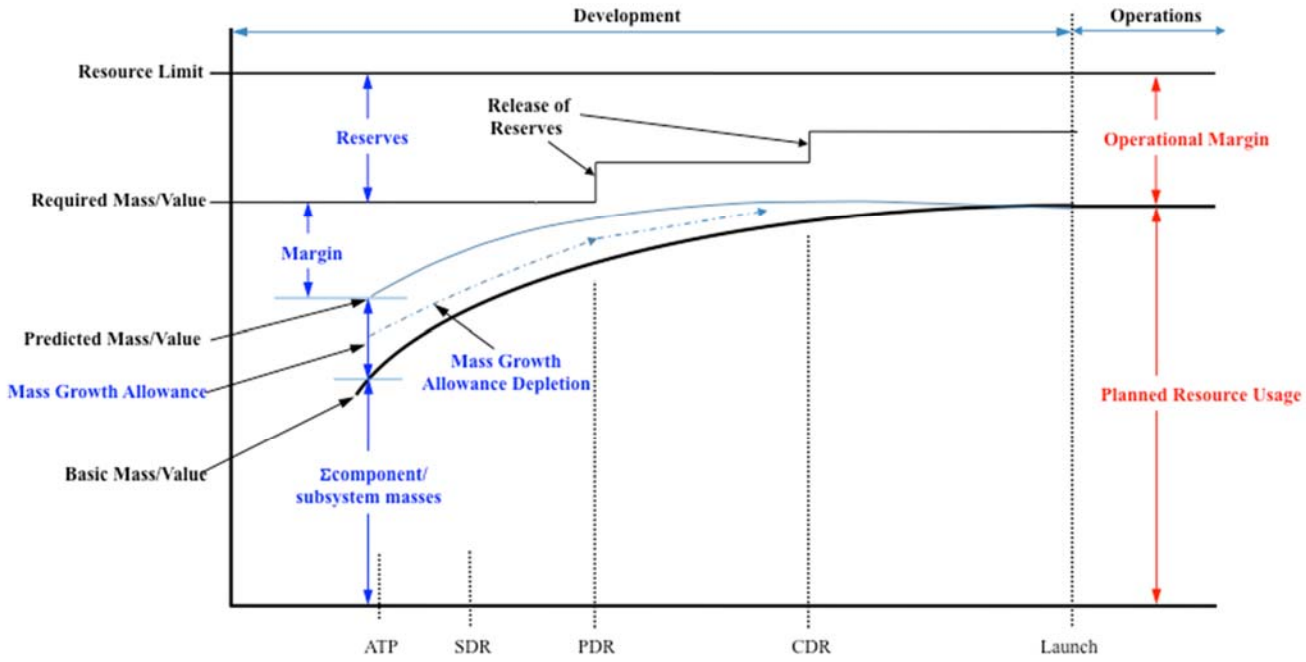


Figure 7.8-2 Mass Margin and MGA Release through the Project Life Cycle

Table 7.8-2 below provides guidance on the sum of mass margin and MGA by project milestone and represents the percentage to add to the predicted value such that the predicted plus margin equals the required mass. As a minimum, the sum of the mass margin and growth allowance should be as specified in the table. Projects are strongly encouraged to identify mass growth allowances separate from mass margin especially for projects with greater complexity and lower acceptable risk. The values below are consistent with approaches taken by many NASA Centers and are supported by studies of previous NASA missions.[4]

Table 7.8-2 Mass Margins Plus MGA

MCR	SRR/SDR/MDR	PDR	CDR	TRR
25-40%	25-35%	20-25%	10-15%	5-10%

[2]"Using Historical NASA Cost and Schedule Growth to Set Future Program and Project Reserve Guidelines," NASA/Aerospace Corp. paper; 2008

MCR – Mission Concept Review; SRR – System Requirements Review; SDR – System Design Review; MDR – Mission Design Review; PDR – Preliminary Design Review, CDR – Critical Design Review; TRR – Test Readiness Review

7.8.3.2 Power and Energy Margin

Margins within the Electrical Power System (EPS) may cover a variety of topics as discussed in *AIAA S-122-2007, Electrical Power Systems for Unmanned Spacecraft*. The most common is margin based on a power or energy balance analysis that seeks to show that the EPS will always provide sufficient power to the loads over the mission life. A worst-case analysis of power or energy consumption throughout the system should be conducted by generating a set of specific operational or orbital scenarios typically called Design Reference Cases (DRCs) to show that sufficient energy is generated or provided in all cases. Table 7.8-3 below provides guidance on power and energy margins by project milestone and represents the percentage to add to the predicted value such that the predicted plus margin equals the required power.

Table 7.8-3 Power and Energy Margins

MCR	SRR/SDR/MDR	PDR	CDR	Launch
25-30%	20-25%	15-20%	15%	10%

MCR – Mission Concept Review; SRR – System Requirements Review; SDR – System Design Review; MDR – Mission Design Review; PDR – Preliminary Design Review, CDR – Critical Design Review

7.8.3.3 Other Resources

In addition to mass and power margin, it is important to establish margins early in the project life cycle for other technical parameters or resources. The technical parameters should be based on the specific design aspects of the project. The list below provides examples of technical parameters that are common to many projects.

- Propellant
- Pointing knowledge
- Pointing accuracy
- Control stability
- Data throughput
- Computer memory
- Data storage
- RF link margin
- Temperature
- Heat rejection capacity
- Response time
- Telemetry, command hardware channels
- Reliability
- Availability
- Torque/force
- Battery cycles/life

7.8.4 General Considerations

Margins that are greater than those shown above may be required depending on project-specific circumstances. For example, highly complex mission and/or system designs, development of low TRL technology, uncertainty of heritage designs, tight performance margins, low budget reserves, and tight schedule margins might be reasons to require higher than the margin plus growth allowance described above. Likewise, margins and growth allowances less than those shown may be acceptable in certain cases. For example, a decreased margin and/or growth allowance may be possible when reusing a known system design with a heritage payload in a mission application and environment previously flown; or in other circumstances where the unknown factors are fewer and/or mature hardware is, by project policy, not to be changed; or where there are ample margins in other technical and programmatic resources.

All growth allowance guidelines assume an average level of uncertainty; it is necessary to adjust growth allowance upward for items with higher uncertainty and downward for items with lower uncertainty. Another approach is to allocate increased dollar reserves to offset lower margins in some areas, e.g., technical performance or unknown development schedules. In order not to over-budget, growth allowance may be applied individually to portions of the system and then summed to define the system growth allowance.

7.8.5 Margin Management Plan (Technical Metrics Plan)

Each project should define and manage margins and growth allowances for mass through the development of a margin management plan or sometimes call a technical metrics plan (see Section 6.7.2.6). Typically mass has its own control plan but for smaller projects may be a part of the margin management plan that covers all project resources or technical metrics. Developing Technical Performance Measures (TPMs) involves defining the necessary quantitative analysis for each TPM or resource to be tracked. It is also advised that threshold values be established to distinguish **GREEN** (high probability of program/project compliance – Action: *none*), **YELLOW** (moderate risk of program/project compliance – Action: *watch*), and **RED** (high risk of program/project non-compliance – Action: *immediate mitigation plan is necessary*).

In some cases, projects use the term margin to cover the intended purpose of both margin and growth allowance. The key is that the project understands where growth in a resource may occur and develops an approach to control its growth to within acceptable limits throughout the project life cycle. The plan formulates a margin and growth allowance control approach based on the critical parameters that need to be controlled and implemented throughout the project life cycle to increase the chances of meeting the project's functional, performance, and/or safety requirements. It is important to note that a plan may be a single page document developed in Word or Excel or a detailed document that describes the roles and responsibility of the project manager, chief engineer or lead systems engineer, and subsystem leads in the management and allocation of the margin and growth allowance throughout the project's life cycle. A project may choose for the project manager to hold most of the margin at their level while allocating the remainder to the chief engineer or lead systems engineer to manager. Other projects may allocate the margin more evenly between the project manager, chief engineer or lead systems engineer, and subsystem leads with the subsystem leads also managing the growth allowance. Each project should determine the level of detail appropriate based on the project's risk posture, complexity, and cost. The plan should be developed very early in the project life cycle, typically Phase A, and be implemented throughout life-cycle Phases B, C, and D, and in some cases Phase E.

7.8.6 Additional Reading and References

The documents listed below provide good background and basis for the values shown above for mass and power but also provide guidance on margins for additional technical parameters or resources. The documents can be found on the NASA Technical Standards or NASA Engineering Network (NEN) Web sites.

Document Number	Document Title
GSFC-STD-1000	GSFC Gold Rules
D-17868	JPL Design Principles
APR 8070.2	Class D Design and Environmental Test Requirements
AIAA S-120-2006	Mass Properties Control for Space Systems
AIAA S-122-2007	Electrical Power Systems for Unmanned Spacecraft
NASA/Aerospace Corp. paper; 2008	Using Historical NASA Cost and Schedule Growth to Set Future Program and Project Reserve Guidelines
NASA/Aerospace Corp. presentation; 2008	An Assessment of the Inherent Optimism in Early Conceptual Designs and its Effect on Cost and Schedule Growth
NASA Cost Symposium; 2014	NASA Mass Growth Analysis - Spacecraft & Subsystems

7.9 Human Systems Integration (HSI) in the SE Process

Human Systems Integration (HSI) is an "interdisciplinary and comprehensive management and technical process that focuses on the integration of human capabilities and limitations into the system acquisition and development processes to enhance human system design, reduce life-cycle ownership cost, and optimize total system performance." (Source: NPR 7123.1.)

The goal of HSI in SE is to balance total system safety and effectiveness and to ensure mission success through iterative attention to efficient interaction of hardware and software design with the total system's most critical, versatile, and variable element: the human. HSI is a set of process activities that ensure (1) the physiological, cognitive, and social characteristics of personnel are addressed in systems development; (2) the systems design supports personnel and includes personnel in an integrated perspective on total system performance, reliability, and safety; and (3) system designs are standardized and consistent across all products HSI supports, in areas such as user interfaces, procedures, and training.

HSI activities include both management and technical processes that work within systems engineering processes and methodologies to ensure successful human systems integration. The approach is interdisciplinary and comprehensive, applying to both management and technical processes throughout the entire product life cycle. HSI is applied to the system design and development processes, the system production and delivery of end product(s), all operations phases, and decommissioning of the end product(s).

Inherent to the rationale of emphasizing HSI in systems engineering is accepting that all engineering is performed to fulfill human needs and accomplish human objectives. Personnel are inherent to the success of any system; i.e., every system includes personnel who use the system and help the system fulfill its objectives. It is critical to consider human users, maintainers, and operators as key parts of the system. Humans bring unique capabilities to any project; e.g., creative thinking, an ability to understand the big picture of the mission, complex communication ability, etc. Humans are the most resilient part of any system and can adapt the system if even remotely possible. At the same time, humans are the most unreliable part of any system, given the inherent limitations of human performance. Acknowledgement of these limitations and capabilities, in the form of early planning and system design, greatly enhance the chance of mission success. Often a human interacting with a system is the 'last line of defense' in maintaining a system's effectiveness with the human being ultimately accountable and responsible for mission success. However, humans have many limitations such as memory (declarative, retrospective, and prospective), vigilance over periods of time, fatigue, social and biological needs. A computer chip does not need to feel that its contribution is meaningful, a human does. Training can be one method of dealing with these limitations, but for training to have value, the human should remember it. Training may need to be refreshed in some way and a comprehensive system perspective would account for this; e.g., with built-in "Help" menus. If human capabilities are relied upon, but human limitations are not sufficiently addressed in design, then the human component of the system is more likely to fail, putting the system's mission performance at risk.

HSI relies on four key concepts to facilitate an effective program. First is the recognition that systems comprise hardware, software, and humans, all of which interact and operate within an

environment. Secondly, human interactions that need to be considered include all personnel that interface with a system; i.e., the end users (pilots, crewmembers), maintainers, ground controllers, logistics personnel, etc. Thirdly, successful HSI depends on the integration and collaboration of all the HSI domains. (See Section 7.9.1.) Finally, HSI should be established early in the design phase of systems and applied iteratively throughout the life cycle of system design, development, and operations.

7.9.1 Integrating Across HSI Domains

Each HSI domain is a discipline expertise (including stakeholders and technical experts) that contributes to design decisions. In order for HSI to optimize total system performance (i.e., human + hardware + software), the appropriate HSI domains should be engaged throughout the system life cycle. As Figure 7.9-1 illustrates, each HSI domain has the potential to affect and interact with the other domains, making it critical to execute an integrated discipline approach. Human Factors Engineering (HFE) is the central domain, in that it is responsible for characterizing human capabilities and constraints and for applying knowledge of these to engineered hardware/software systems' design. Recommendations by HFE influence mission success and operations costs associated with the other domains.

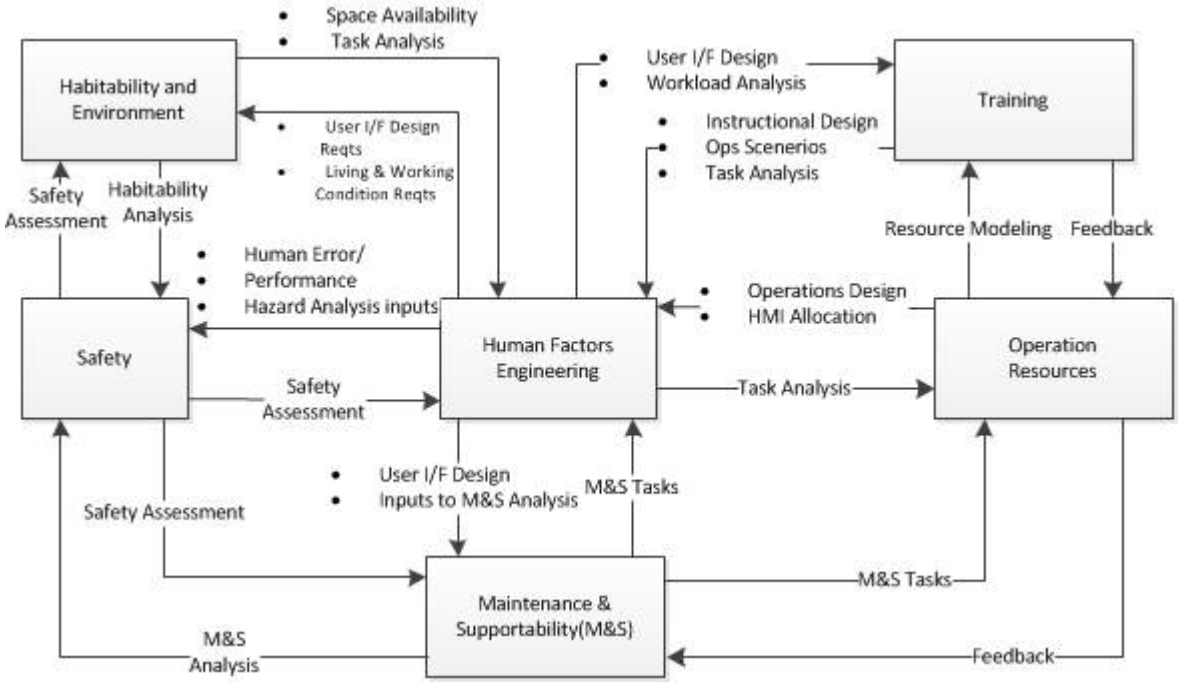


Figure 7.9-1 Notional HSI Domain Interaction

NPR 7123.1, NASA Systems Engineering Processes and Requirements, requires programs and projects to develop an HSI Plan that shows how HSI will be conducted. (See Section 7.9.5 and appendix R of this document.) An HSI team, comprising multiple domain discipline experts and/or representatives, may be established to fulfill this requirement in coordination with project management and SE personnel. HSI domain expertise may reside at multiple NASA Centers and an HSI team need not be physically colocated. Each program/project will have to tailor its own

list of applicable domains, but tailoring should involve Agency Technical Authority input to ensure thorough and proper implementation of HSI.

NASA HSI domains are listed in Table 7.9-1. HSI personnel with integrated domain oversight are best positioned to implement HSI processes and practices. HSI personnel who are skilled at integrating program/project HSI inputs from across the individual HSI domains may also have deep expertise in one or more of the individual domains. But the HSI integrator's primary task is crosscutting integration across domain inputs; i.e., the HSI integrator does not replace domain expertise. Functional implementation of HSI is based on regular, frequent communication, coordination, and integration across the HSI domains providing human-systems expertise.

Table 7.9-1 NASA HSI Domains

Domain	Definition	Examples of Expertise
Human Factors Engineering (HFE)	Design to optimize human well-being and overall system safety and performance by emphasizing human capabilities and limitations as they impact and are impacted by system design across mission environments and conditions (nominal, contingency, and emergency) to support robust integration of all humans interacting with a system throughout its life cycle. HFE solutions are guided by three principles: system demands shall be compatible with human capabilities and limitations; systems shall enable the utilization of human capabilities in non-routine and unpredicted situations; and systems shall tolerate and recover from human errors.	Crew Workload and Usability, Human-in-the-Loop Evaluation, Human Error Analysis, Human Interface & Systems Design
Operations Resources	The considerations and resources required for operations planning and execution. This includes operability and human effectiveness for flight and ground crews to drive system design and development phases, as well as trades for function allocation, automation, and autonomy.	Operations process design for both ground and flight crew, Human/machine resource allocation, Mission Operations, Resource modeling, Flight Operations
Maintainability and Supportability	Design to simplify maintenance and optimize human resources, spares, consumables, and logistics, which is essential due to limited time, access, and distance for space missions.	Inflight Maintenance and Housekeeping, Ground Maintenance and Assembly, Sustainability and Logistics
Habitability and Environment	External and internal environment considerations for human habitat and exposure to natural environment including factors of living and working conditions necessary to sustain the morale, safety, health, and performance of the user population, which directly affect personnel effectiveness.	Environmental Health, Radiation Health, Toxicology, Nutrition, Acoustics, Architecture Crew Health and Countermeasures, EVA Physiology
Safety	Safety factors ensure the execution of mission activities with minimal risk to personnel. Mission success includes returning crew following completion of mission objectives and maintaining the safety of ground personnel.	Safety, Reliability, Quality Assurance

Domain	Definition	Examples of Expertise
Training	The instruction and resources that are required to provide personnel with requisite knowledge, skills, and abilities to properly operate, maintain, and support the system.	Instructional Design, Training Facility Development

7.9.2 HSI Roles and Responsibilities

On any specific program or project, the parties deemed responsible for providing integrated HSI input are—in conjunction with program/project management and systems engineering—responsible for implementing HSI processes throughout the program's/project's life cycle. As an example, ensuring effective integrated HSI domain engagement in crewed space system design can reduce in-flight risk to human health and performance. For robotic space missions, the human operators are Earth-based, but HSI processes are no less critical to total system performance and survivability; HSI is essential to ensuring the design of operable flight and operations systems, clearly allocating functions between humans and systems, maximizing mission return, and reducing the likelihood and impact of human error--for which (in the case of robotic missions) there is no crew onboard to mitigate resulting behavior by taking onboard corrective action.

7.9.2.1 Program/Project Management

Based on the intent of NPR 7123.1, every program is expected to perform HSI; i.e., for all program types, the program/project manager and systems engineer should integrate HSI into the SE process throughout the program life cycle to positively influence total system effectiveness and cost. NPR 7123.1 states, "Hardware, software, and human systems integration considerations should be assessed in all aspects of these processes" (referring to systems engineering processes). The program/project manager may be involved more heavily with the HSI process early in the system life cycle to ensure that appropriate HSI disciplines are represented, that an HSI Plan is developed consistent with the program/project SEMP. For human-rated space systems, it is the responsibility of the program/project manager to form an HSI team before SRR (per NPR 8705.2, Human-Rating Requirements for Space Systems). For human-rated programs/projects, the program/project management relies on the HSI team throughout the system life cycle to keep the design focused on stakeholder and user expectations, to elevate issues, and to document formal acceptance or lack of acceptance regarding HSI deliverables.

For programs/projects not requiring NPR 8705.2 human-rating, it is the responsibility of the program/project manager to determine who (individual or team) is responsible for overall program/project HSI, for developing the HSI Plan, and for HSI implementation and results.

7.9.2.2 HSI Team

Formation of an HSI team may be required (by NPR 8705.2 for human systems) or it may be deemed by the program/project manager to be the most efficient and effective programmatic approach to HSI implementation. Required for human-rated space flight programs by NPR 8705.2, Human-Rating Requirements for Space Systems, the formation of an HSI team is a recommended practice for any program/project of sufficient size requiring engagement of multiple HSI domain discipline skills and expertise. The HSI team may serve as the

representative body for HSI and human-centered design implementation, providing recommendations to any and all oversight boards and panels and/or directly to the program/project manager. The HSI team may be given the authority to elevate issues for resolution tor program/project management and to document formal acceptance or lack of acceptance regarding deliverables. If established, a thorough HSI team should include HSI practitioners and integrators, system developers, and system operators, HSI domain experts, and other stakeholders. To avoid confusion, the roles and responsibilities of an HSI team and the members that comprise it should be clearly documented in the program- / project-specific HSI Plan. If an HSI team is not the desired HSI implementation, the program / project manager needs to designate the party or parties responsible for implementing HSI no later than Phase A.

Even if the program/project management elects not to form an HSI team, the approach to performing HSI and delivering HSI products is to be documented in the HSI Plan. The HSI team (or alternate implementation) manages HSI domain interaction with system designers between milestone reviews, provides HSI guidance and expertise, and ensures that Human-Centered Design (HCD) issues are identified early to minimize cost and schedule impacts. Good insight into design progress between scheduled program/project milestones facilitates review of applicable materials at each milestone, reinforcing the concept of efficient inclusion of HSI throughout the program/project life cycle as part of a HCD process.

For successful HSI implementation, an HSI team (or alternate responsible party) includes or has access to sufficient depth and breadth of HSI domain discipline technical expertise to implement an HSI Plan and to meet HSI objectives.

7.9.3 Mapping HSI into the SE Engine

HSI processes are integral to effective systems engineering. HSI processes, best practices, and tools fit seamlessly into the NASA standard SE framework, such that by implementing the SE engine as shown in Figure 2.1-1, HSI processes are executed. A mapping of HSI topics into the NASA SE processes is shown in Table 7.9-2. HSI consideration throughout SE life-cycle activities begins by ensuring that stakeholders for the full life cycle are identified during program/project formulation and that stakeholder needs are identified early and validated throughout the life cycle. HSI's focus on incorporating operations into design decisions begins in Pre-Phase A by considering the human as a part of total system performance. Considering HSI early in the life cycle provides a foundation for full life cycle HSI diligence that can produce human/system interaction metrics supporting cost-efficient training, increased system reliability, easier and more efficient system maintenance, and increased safety and survivability.

Table 7.9-2 Mapping HSI into the SE Engine

System Design Processes	HSI Emphasis
Requirements Definition Processes Stakeholder Expectations Def. (1) Tech. Req'ts Def. (2)	Functional allocation between and among systems and humans, define roles and responsibilities, develop requirements, baseline ConOps
Technical Solution Definition Processes Logical Decomposition (3) Design Solution Definition (4)	Functional allocation (during decomposition), ConOps and ops goals, iterative human-centered design, design prototyping for human-in-the-loop evaluation

Product Realization Processes	HSI Emphasis
Design Realization Processes Product Implementation (5) Product Integration (6)	Validate design for all human-systems interactions as elements are integrated
Evaluation Processes Product Verification (7)Product Validation (8)	Human-in-the-loop testing, validation to ConOps
Product Transition Process (9)	Prepare for Operations: training, simulations, procedures
Technical Management Processes	**HSI Emphasis**
Technical Planning Processes (10)	Life-cycle cost management
Technical Control Processes Requirements (11), I/F (12), Risk (13), CM (14), Tech Data (15)	HSI participation in management processes, as required
Technical Assessment Process (16)	HSI products, entrance, and exit criteria for milestone reviews; TPM examples
Technical Decision Analysis Process (17)	Human-centered design, HSI domain participation

HSI is a crosscutting technical management process that is applied throughout the system life cycle by considering all HSI domains at each phase and engaging the appropriate expertise. By systematically infusing information from past designs, operational use, and user feedback, optimal designs are created that are validated against the concept of operations and mission performance goals at each milestone.

Human-centered design approaches developed and applied by HSI practitioners are central to an HSI approach and program-/project-specific metrics (e.g., personnel and training quantification, workload, turnaround time, etc.) are used to track HSI goals and requirements compliance. Human-Centered Design (HCD) is an approach to interactive system development that focuses on making systems usable by ensuring that the needs, abilities, and limitations of the human user are met. HCD is a multi-disciplinary activity that involves a range of skills and stakeholders that collaborate on design. Most importantly, HCD is an iterative activity that intentionally uses data gathered from users and evaluations to inform designs. The benefits of the HCD approach can be realized in terms of cost control, mission success, and customer satisfaction. The following section provides more information on HCD.

7.9.4 HSI Activities

The full scope of HSI implementation includes the following sequential and iterative activities:

- Function allocation between hardware/software systems and humans;
- Concept of operations (which should address off-nominal scenarios and the ConOps for training, maintenance, logistics, and sustainment as well);
- Requirements interpretation;

- Task and user analyses;
- Allocation of roles and responsibilities among humans engaged with the system;
- Iterative conceptual design (e.g. prototyping, modeling, tradeoffs);
- Human-In-The-Loop (HITL) testing;
- Model-based performance assessment;
- Support verification and validation of products and design solutions;
- Monitoring for appropriate operational human physiological and cognitive support; and
- Operational data collection and lessons learned.

These activities are equally relevant for aeronautic, robotic science missions, and human spaceflight. *NASA/SP-2010-3407, Human Integration Design Handbook*, Section 3.3, "Application of the HIDH to System Design and Development," describes the application of these processes to each NASA systems engineering life-cycle phase.

These processes are consistent with the Human-Centered Design (HCD) process required by *NASA-STD-3001, Volume 2: Space Flight Human-System Standard—Human Factors, Habitability, & Environmental Health*, Paragraph 3.5, "Human-Centered Design Process." More and/or supporting detail can also be found in *JSC-65995, Commercial Human Systems Integration Processes (CHSIP)* and NASA/TM-2008-215126/Volume II (NESC-RP-06-108/05-173-E/
Part 2), Design Development Test and Evaluation (DDT&E) Considerations for Safe and Reliable Human-Rated Spacecraft Systems, Section 11, "Human Factors Engineering."

Key contributions from HSI include life-cycle HSI requirements integration and management; i.e., requirements development, functional flow to every level of subsystem breakdown, requirements interpretation, and ultimately, requirements verification. HSI should reflect a "top-down" process that starts with the program's/project's high-level mission and goals. These are divided into the functions necessary to achieve the goals, which are then allocated to human and system resources. Functions are broken down further into human and/or system tasks and analyzed to identify the requirements for effective and safe task performance to support both the humans' capabilities / limitations and overall integrated system performance. Human-allocated tasks are arranged into work activities to be performed by individual personnel and/or teams. The detailed design of the user interfaces (e.g., alarms, displays, and controls), procedures, and training represent the "bottom" of the HSI's top-down process. After participating in HSI requirements development, HSI participates in design activities to ensure that documented human system requirements have been thoroughly communicated to all systems design team members, appropriately flowed to subsystems, and effectively implemented. Requirements interpretations are provided when needed to ensure the intent of Agency human-system standards (e.g., NASA-STD-3001 or the FAA's Human Factors Design Standards) is being translated into design implementation. Experts from the HSI domains and experienced HSI integrators are best suited to work directly with program-/project-level requirements developers and system designers to develop and implement HSI.

As system designs matures, the effectiveness of HSI implementation should be validated through prototyping, human-in-the-loop validation, and/or analyses in areas such as radiation, anthropometry/biomechanics, environmental factors (e.g. air, water, toxicity, O2, CO2, humidity, temperature), lighting, task analyses, human error analyses, in-flight and ground maintenance, and ground support equipment human engineering. HSI contributes to integrated development test planning and execution, ensuring that human-system interfaces and interactions are part of system validation. Examples of HSI development phase testing may include Human-In-The-Loop (HITL) evaluations, Environmental Control and Life Support Systems (ECLSS) test-bed human testing, display format evaluations, workload/usability/handling qualities evaluations, suit-vehicle integrated HITL evaluation, robotic mission flight-control workload and off-nominal operations evaluations, vehicle (crewed or robotic) operability assessments, and ground operations stress and workload evaluations.

HSI personnel participate in design iteration activities to reduce ground and flight personnel numbers, ground and flight training needs, simplify maintenance and logistics, avoid mishaps (design or operations), and minimize system design rework. This is accomplished by engaging all domains early and often in the life cycle and through analysis of alternatives. Without proper tradeoff assessments and resulting design iteration, skipping ahead quickly during the design phase can create "technical debt" that has to be "paid for" during production, test, and operations phases, as shown in Figure 7.9-2. Human modeling may be used early in the life cycle. As the design progresses in maturity, more in-depth analyses are performed on an integrated system using human-in-the-loop tests.

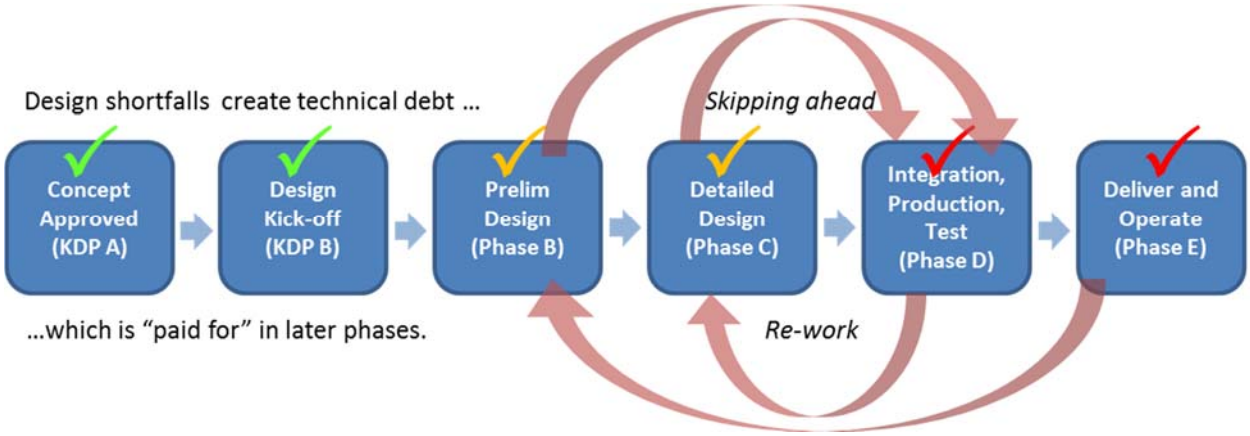


Figure 7.9-2 HSI Goal: Reduce Rework

In addition to being engaged in early design activities, HSI participates in product realization (see Figure 2.1-1) by performing tasks such as the following:

- Participation in program analysis and requirement verification cycles: supporting design analysis cycles and verification analysis cycles by conducting HSI analyses and trade studies that are necessary to refine the system design and to verify that the system meets baseline and successive requirements.
- Participation in major reviews such as design milestone and system safety reviews to ensure meeting the programmatic and systems engineering entry and success criteria of these major reviews. This may include providing presentation materials that explain the maturity of the

design as related to human-system integration. Reviewing deliverables at each program milestone ensures iterative and adequate HSI design considerations are taking place.

- Acceptance testing of each set of deliverable hardware, including human-related functional testing, hardware inspections, sharp-edges testing, and other acceptance tests.
- Support for NASA institutional Flight Readiness Reviews (FRR) and Airworthiness Safety Review Board (ASRB) flight release processes for documenting the readiness of flight system and mission preparations as related to HSI requirements and processes.
- Participation in validation and verification activities throughout the vehicle development life cycle to ensure that proposed design solutions will meet not only program mission requirements but also evaluate the effectiveness of the HSI effort.

HSI participates in formal verification of HSI requirements directly and/or contributes significantly to verification planning, assessment, and closure. In addition, HSI performs product validation of vehicle design and operations. HSI personnel should participate in program/project validation events, including requirement analyses, demonstrations, test flights, simulations, analyses, and human-in-the-loop evaluations.

7.9.5 Products and Tools

A program's/project's HSI approach should be tailored to include the use of various products and tools as appropriate. Note, however, that HSI data is often integrated with other data in a standard product of the program/project and not uniquely an HSI product. Any product could have HSI implications; human considerations naturally occur as part of effective capability-based systems engineering.

7.9.5.1 HSI Plan

An HSI Plan is the focal HSI product for a program/project since it serves as the roadmap for HSI implementation. An outline for a program/project HSI plan is in appendix R of this document. As long as the intent and content of an HSI Plan are captured, the HSI Plan may be a stand-alone systems engineering product, or it may be incorporated into a particular program's/project's SEMP or program/project plan. (A stand-alone SEMP and HSI Plan are recommended for programs and large projects; having the HSI Plan incorporated into the SEMP or program / project plan may be appropriate for small projects.) If the plan is stand-alone, the parties responsible for systems engineering and HSI should ensure that the HSI Plan is aligned with the SEMP. The HSI Plan defines how human system considerations are integrated into the full systems engineering design, verification, and validation life cycle. The HSI Plan is a living document with updates to be made at significant program/project milestones. NPR 7123.1 states that the HSI Plan is first developed to support SRR, with updates required at SDR, MDR/PDR, and CDR that document the implementation of an HSI design approach to the system and its mission and that demonstrate how the design accommodates human capabilities and limitations. The HSI Plan should indicate how HSI will document issues, risks, and their mitigation as they are worked during the life cycle. By developing and executing the HSI Plan, the PM expends the effort—in conjunction with designated parties responsible for the program's/project's systems engineering and HSI implementation—to integrate, capture, and track HSI metrics throughout the life cycle of the program to increase safety, total system performance, and mission success.

An HSI checklist or electronic scorecard may be used to support implementation of a program- / project-specific HSI Plan. The checklist can serve as a field guide of HSI considerations at each phase of the systems engineering life cycle and to help measure design compliance with HSI requirements. At each SE milestone, an evaluator might use an electronic scorecard to fill in answers to specific questions that track progress on requirement compliance. These tools can help show whether or not a design is on track for each specific life-cycle milestone.

7.9.5.2 HSI Requirements

HSI requirements are an important HSI product. Requirements are the ultimate tool for impacting system design and performance, but they often also have cost and schedule implications. HSI requirements ensure that the human is adequately considered during system design. HSI requirements are developed, integrated, interpreted, and verified with support from parties responsible for HSI, from systems engineering personnel, and from discipline experts in each HSI domain.

7.9.5.3 Other HSI Products

Additional HSI products may be specific to each program/project and to each domain. Products such as an acoustics noise control plan, a task analysis, human-in-the-loop verification plans, usability analysis results, radiation shielding models, habitability assessments, system maintenance plans, and display standards are examples of domain-specific HSI products. These products contain essential data to ensure that human capabilities and limitations are adequately factored into design of the total system.

7.9.5.4 HSI Tools

HSI tools are available to contribute towards effective HSI implementation. As with products, there are tools that are specific to HSI as well as systems engineering tools that can be used for HSI purposes. HSI components for Model-Based Systems Engineering (MBSE) are currently at low levels of maturity, but this is an area where rapid improvement is anticipated.

DoD HSI Tool Resources

Additional information on available HSI tools may be found in the Department of Defense (DoD) Defense Technical Information Center's (DTIC's) Directory of Design Support Methods (DDSM) (Defense Technical Information Center, 2007). According to the preface of this resource: "The DDSM provides an annotated directory of human systems integration (HSI) design support tools and techniques that have been developed by the DoD, NASA, FAA, NATO countries, academia, and private industry….The DDSM contains references to design tools or techniques that are currently available or under development. New records continue to be added as new human systems tools and techniques are developed."

http://www.dtic.mil/dtic/tr/fulltext/u2/a437106.pdf

7.9.5.4.1 HSI Key Performance Metrics

HSI key performance metrics that can be quantified and tracked throughout system development and that predict and characterize total human-plus-system performance outcomes for the operations phase are an important HSI tool. These metrics translate into early and ongoing cost

evaluation of the full life cycle of a project. Metrics allow HSI efforts to yield quantifiable and measurable impacts to system design. By tracking data such as time required for training on a system, time required for system operation and maintenance, and number of HSI issues documented in the HSI Plan, engineers can identify key areas for HSI investment. In addition, tracking the status of the HSI metrics helps to determine the program/project HSI maturity level and effectiveness as the program/project moves forward.

Examples of HSI metrics include the following:

- Crew time or task efficiency (i.e., measured operational performance versus expected);
- Training time across design alternatives trade studies;
- Total numbers of operations personnel and skill sets required;
- Numbers of human interactions with major systems and subsystems; and
- Estimated life-cycle cost.

7.9.5.4.2 Other Tools for HSI

Section 4.3.2.2 describes use of SE tools such as functional flow block diagrams (FFBD), N-squared (N2) diagrams, and timeline analysis. Each of these tools can be applied specifically for HSI by focusing on an area of human performance or function. Output of the tools is the same as when used for a strictly hardware system, only the HSI output would be that of the human system and/or human interfaces. Outputs of tools common to systems and/or humans might be more readily integrated to serve as measures of total system performance.

7.9.6 HSI and Life-Cycle Cost Reduction

A well-executed implementation of HSI in a program can produce significant cost-avoidance. Effectively applying HSI processes reduces life-cycle cost by bringing operational experience and goals to light during design and development.

In NASA systems (human space flight, robotic mission, and aeronautics), HSI should not only focus on the interaction between the operators (such as the aircraft or spacecraft flight crew) and the hardware/software systems, but on all ground and flight personnel that interact with the system throughout its lifetime. In fact, the bigger savings in total system development, deployment, and operational management are likely to come from careful consideration of the human skills and manpower required for total system logistics, maintenance, and mission operations. In an effective HSI program, the personnel and infrastructures required to make and keep the system fully operational are considered as part of the system and consideration of their operational needs and expectations are addressed during development through HSI metrics tracking, analysis, human-in-the-loop testing, and evaluation. The accuracy of any quantification of total human / system performance is reliant on a complete accounting of "what makes the system work." Efficient total human/system performance is the goal in HSI and efficiency goals should be established early in HSI implementation and tracked through the life cycle. Many human-machine designs have been able to make the system work, but at considerable waste of personnel and training resources and with vulnerability to human error.

Rework of any system to retrofit for human/system operational efficiencies can be extremely expensive when identified late in the design/production cycle. Delays from poor design of standard operational processes such as maintenance activities can increase costs. Poor systems and human-systems interface designs can lead to damage, which can increase cost and schedule as repairs are made or, if the damage is unnoticed, decrease mission success or increase mission risk.

Later in a system's life cycle, there is no guarantee that retrofitting will be sufficiently affordable to even become a consideration. Figure 2.5-3 indicates the steep rise in "cost to extract defects;" i.e., to change a system design as the life cycle progresses. This indicates that there is significant potential for cost efficiency in iterating design and stakeholder reviews while the program/project is still in its earliest phases of system development. Setting targets for human/system operational efficiencies from the outset of design coupled with HSI diligence throughout the life cycle helps to avoid unexpected costs for personnel expense or retrofit in the operations phase of the life cycle. The intent of comprehensive, life-cycle HSI implementation is to ensure and validate that the design meets stakeholder needs, and that for all life-cycle phases, there is an integrated and balanced approach to design and implementation across hardware, software, and human elements that comprise the total system.

Cost-effectiveness can be achieved by an HSI approach to design, management, and systems engineering. As an example, a new exploration program may require individuals to control the precise movements of remote robotic systems, or an unmanned aviation system may require accurate control of vehicle positioning and surveillance, either requiring extensive training to operate the system. HSI analyses of remote operation tasks and workstation interfaces may indicate an alternative design solution can be chosen that reduces the time to complete tasks. As another example, after several design iterations involving human-in-the-loop evaluation of prototype concepts, a more cost-effective approach to science data collection might be demonstrated from a less complex and risky system solution than originally presumed. Such increases in effective work output may require additional up-front cost for development iteration but could demonstrably manifest savings in total life-cycle costs. HSI needs to be considered within the trade space and as a tool to help establish the trade space. The potential of HSI is that a program's/project's overall life-cycle cost is reduced and final system designs are less complex for personnel to use and entail less risk in meeting mission objectives.

HSI can be viewed as "think about the end at the beginning." The ultimate goal of any system is to perform safely, efficiently, and accurately while meeting expectations. Focusing on stakeholder needs and continually assessing the design against those needs, by validation against the ConOps for example, opens the design space to incorporate humans at the same level as hardware and software.

> You can use an *eraser* on the drafting table or a *sledge hammer* on the construction site.
>
> Frank Lloyd Wright

7.9.7 NASA HSI Body of Knowledge

The documents in Table 7.9-3 contribute towards HSI implementation at NASA. Specific HSI content in each document is noted.

Table 7.9-3 NASA Documents with HSI Content

Document	HSI Content
NASA-STD-3001, NASA Space Flight Human System Standard	NASA Office of the Chief Health and Medical Officer (OCHMO) mandatory standard for NASA human space flight programs. Establishes Agencywide requirements that minimize health and performance risks for flight crew in human space flight programs. Includes requirement [V2 3005] mandating that human space flight programs establish and execute a human-centered design process.
NPR 8705.2, Human-Rating Requirements for Space Systems	Processes, procedures, and requirements necessary to produce human-rated space systems that protect the safety of crew members and passengers on NASA space missions. For programs that require human rating per this NPR, paragraph 2.3.8 requires the space flight program to form an HSI team before SRR.
NPR 7123.1, NASA Systems Engineering Processes and Requirements	Appendix A includes definition of HSI. Appendix G, Life-Cycle and Technical Review Entrance and Success Criteria, includes an HSI Plan.
NASA/SP-2010-3407, Human Integration Design Handbook (HIDH)	Guidance for the crew health, habitability, environment, and human factors design of all NASA human space flight systems.
NASA/TP-2014-218556, Human Integration Design Processes (HIDP)	HSI design processes, including methodologies and best practices that NASA has used to meet human systems and human-rating requirements for developing crewed spacecraft. HIDP content is framed around human-centered design methodologies and processes.
NASA/SP-2014-3705, NASA Space Flight Program and Project Management Handbook (companion to NPR 7120.5, NASA Space Flight Program and Project Management Requirements)	Contains context, detail, rationale, and guidance that supplements and enhances the implementation of space flight programs and projects, including a HSI Plan.
NPR 8900.1, NASA Health and Medical Requirements for Human Space Exploration	Establishes health and medical requirements for human space flight and the responsibilities for their implementation including health and medical, human performance, habitability, and environmental standards; and sponsorship of health-related and clinical research.
NPR 7120.11, NASA Health and Medical Technical Authority (HMTA) Implementation	Implements HMTA responsibilities to assure that Agency health and medical policy, procedural requirements, and standards are addressed in program/project management when applicable and appropriate.

Document	HSI Content
NASA/SP-2015-3709, Human Systems Integration Practitioners Guide	Aids the HSI practitioner engaged in a program or project and serves as a knowledge base to allow the practitioner to step into an HSI lead or team member role for NASA missions. Additionally, this guide is written to address the role of HSI in the program/project management and systems engineering communities and aid their understanding of the value added by incorporating good HSI practices into their programs and projects.

8.0 Special Topics

The articles in this chapter represent topics that are of special interest, may be relatively new to the Agency or may be new methods that can provide benefit. These topics represent useful approaches to system engineering and the sections below provide information on the application of statistical engineering and Model-Based System Engineering (MBSE) on programs, projects, or activities. As these topics are still emerging in their forms and applications within the Agency, there exists flexibility in how to apply these methods to a particular program, project, or activity. In today's computer-based, data-rich world, the systems engineer needs to deal with statistical information and model-based engineering approaches employed by various engineering disciplines. The extent to which statistical engineering and MBSE are applied depends on the judgment of the systems engineer about the benefits to technical, schedule, and cost performance that these approaches provide. The systems engineer should also consider the organizational effects of applying these methods including efficiency and the organization's cultural acceptance.

8.1 Statistical Engineering as a Tool

Statistical engineering is a discipline that integrates engineering disciplines and statistical sciences to solve technical challenges with a quantified level of confidence. The objective is to engineer statistical methods to generate better approaches that benefit organizations through a value-added understanding of uncertainty and ambiguity to achieve research objectives. Literally, it engineers statistical sciences to generate better solutions to large, unstructured problems. (See Hoerl and Snee.) Statistical Engineering supports the effective application of statistical thinking and methods across the research, development, and procurement life cycle resulting in:

- Improved specification of requirements that achieve high-level objectives;
- Faster understanding of system capabilities through accelerated characterization;
- Efficient and effective test programs that minimize test resources; and
- Improved quantification of risk, thereby supporting better decision-making.

For more general information on statistical engineering, see the bibliographical Web site maintained by the American Society for Quality (ASQ), Statistics Division. Within NASA, statistical engineering has been demonstrated on numerous projects that span across NASA's missions of exploration, science, and aeronautics. For more information on NASA applications

of statistical engineering, see the proceedings of the NASA Statistical Engineering Symposium (NSES).

While the application of statistical methods is ubiquitous in engineering, statistical engineering provides a systems perspective that transcends the application of methods in specific disciplines. It may be helpful to consider an analogy to the discipline of mechanical engineering that develops theory and practice for practical application of fundamental sciences such as calculus, physics, and chemistry. In a similar manner, statistical engineering is an approach to build solution approaches from fundamental statistical sciences to generate impactful solutions, particularly to complex, unstructured problems.

Statistical engineering is a systems view of the knowledge sought from a research and development effort. It is based on identifying what we need to know or learn from the physical system that is being developed. NASA's vision and mission are to explore the unknown for the benefit of mankind and to drive advances in science, technology, and exploration to enhance knowledge. This is accomplished through research, analysis, and experimentation to observe and probe systems to find causal relationships between factors and responses. Experimentation is primarily designed to obtain knowledge, understanding, and provide new insights. In essence, NASA projects are initiated to confirm something we believe to be true or to make new discoveries. It can be a common misconception that experiments are conducted to acquire data; rather, a statistical engineering perspective is focused on knowledge, decisions, and impact.

Statistical engineering provides a framework for identifying and accommodating uncertainty and ambiguity in the formulation, planning, execution, analysis, operations, and interpretation of research and development programs. It supports risk-informed decision-making, reliability assessments, probability analyses, probabilistic risk assessment, forecasting and predictions, etc., and ensures technical excellence by efficiently achieving program objectives and quantitatively answering research questions. A statistical engineering perspective focuses on ensuring the integrity of the programmatic and scientific conclusions through the processes and methods employed. Consistently applying statistical engineering principles improves programmatic and technical decision-making, instills greater technical excellence, provides more reliable and predictable outcomes, and ensures more efficient utilization of available resources.

Statistical engineering is closely related to the crosscutting technical management processes of decision analysis (see Section 6.8) in that it provides rigor to the inputs of a formal decision analysis. It ensures that decisions are well-founded and lead to a technically defendable approach to program execution. In addition, statistical engineering is vital to successful technical risk management (see Section 6.4) that combines the probability of an undesired event with the consequence of its occurrence, thereby validating that the safety or performance requirements can be met by using the system specifications requirements defined by the decision analysis. It seeks to ensure a structured process to identify technical and programmatic risks, quantify their magnitude, and link them to programmatic and product consequences. While decision analysis and technical risk management are well-established processes, statistical engineering brings additional rigor to support and defend decision parameters by making their practice more consistent and less idiosyncratic.

Integrating the concepts of statistical engineering early in program formulation and consistently throughout the life cycle is recommended as a best practice. It can support and enhance the system design processes. At formal milestone reviews, infusing a statistical engineering perspective can assist the project in answering fundamental, plain-language questions adapted from the Heilmeier questions (see Shapiro 1994), which are outlined below.

- Program and Project Definition
  - What is the precise objective(s)?
  - Is the objective(s) quantifiable?
  - What are we seeking to learn, or new knowledge sought?
  - How will we know when we have learned it?
  - Is success detectable and measurable?
- Technical Risk Management
  - How well do we need to know the answer(s)?
  - What risk are we willing to accept if we are wrong about our conclusions?
  - What are the consequences if we are wrong?
- Planning and Execution
  - Do the methods support rigorous answers to the stated objectives and risk?
  - Does the allocation of resources reflect support the objectives and risk?
  - Are the resources justifiable and defendable?

While these questions appear straightforward, answering them quantitatively often poses a challenge for a program or project and facilitates substantive discussions that help to refine objectives. Although it is challenging, developing answers to these questions is a role for statistical engineering and enables clear, succinct communication of the project's success criteria throughout the organization and quantitatively supports resource justification to obtain the research objectives. Furthermore, these questions apply recursively through systems and subsystems and throughout the project phases. For effective project leadership, it is accepted that these questions need to be addressed satisfactorily in every phase of a NASA program and project.

In summary, statistical engineering is a complimentary discipline to the systems engineering process. Statistical engineering provides a framework for integrating, linking, and sequencing statistical thinking and tools to improve project performance and for more reliably realizing research and development objectives. Strategically institutionalizing its practice will improve the Agency's ability to achieve its mission.

8.2 Model-Based Systems Engineering

> *"MBSE is part of a long-term trend toward model-centric approaches adopted by other engineering disciplines, including mechanical, electrical and software. In particular, MBSE is expected to replace the document-centric approach that has been practiced by systems engineers in the past and to influence the future practice of systems engineering by being fully integrated into the definition of systems engineering processes."* (Source: INCOSE 2007)

8.2.1 Introduction

Model-Based Systems Engineering (MBSE) is defined as "the formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life-cycle phases." (See INCOSE 2007.)

Systems engineers have used models of various types to help understand, describe, and analyze different aspects of a system. Indeed, we all are using models of a system: within our minds; in drawings, budgets, or equations on paper; or in information that we access or process with computers. One difference between traditional document-centric methods and model-centric methods is that in model-based systems engineering, models are expressed, developed, and matured in a *machine-usable form* external to the engineer.

Traditional practices tend to rely on *multiple, stand-alone* models, resulting in disconnected system representations. These are often discipline-specific models—systems, mechanical, electrical, thermal, etc.—that may be connected by awareness of the engineer, but are disconnected from each other in the sense that they can only be made mutually consistent through acts of human labor. Much of the communication among different engineering teams takes place orally or visually in a discipline-centric viewpoint using a variety of documents that includes human-readable text, diagrams, and spreadsheets. In this approach, the systems engineer can be challenged to ensure consistency among all the disparate models, *especially* as the models are changed over time by their custodians. With disconnected system representations, it can be difficult to get an accurate system-level understanding of the technical baseline.

MBSE includes a paradigm shift from disconnected system representations to systems descriptions in the form of *integrated system models*. MBSE uses formal system models as the preferred way to represent systems, systems engineering activities, and their resulting artifacts, and manage the process of engineering. Because formal models can be subjected to formal tests of completeness, accuracy, and consistency, the integrated system models of MBSE offer an improved way to analyze the system architecture, providing the ability to detect problems earlier in the project life cycle. Formal systems models offer these advantages because they introduce additional rigor and flexibility, because they are both human and computer understandable, and because they are logically verifiable. Additionally, when the system models are integrated by machines, it becomes possible to keep engineering information consistent rapidly.

Integrated systems models help systems engineers manage the many kinds of interrelated information in systems of increasing size and complexity. Systems engineers have always had to capture, in one form or another, information about a system's structures, behaviors, constraints,

and requirements. With the increased existence of standard modeling languages for systems engineering such as OMG SysML, systems engineers can specify and maintain semantically rich relationships among model elements, such as how one component is part of another, how one function depends on another, which requirements specify a component's interfaces, what work package has delivery responsibility for a subsystem, what analysis shows that a performance requirement can be satisfied, etc.

MBSE shifts the locus of authority of the systems descriptions from documents to models. This does not mean eliminating required documents or other traditional systems engineering deliverables. Instead, these artifacts can be increasingly produced automatically from information in the models—the "one source of truth"—ensuring consistency among the artifacts.

The key assumption made by the MBSE approach is that the integrated system model and its representations, or views, describing the system are more capable of describing systems than are documents. There can be a wider variety of views of the system, tailored to the stakeholder interests when MBSE is applied, instead of a standard set of limited documents. There may still be a mix of models and documents generated from the models; some things might be better conveyed in document form, while some others might be better conveyed in models. A benefit of MBSE is the possibility of generating document-formatted reports consistently from the information in various models, as depicted in Figure 8.2-1.

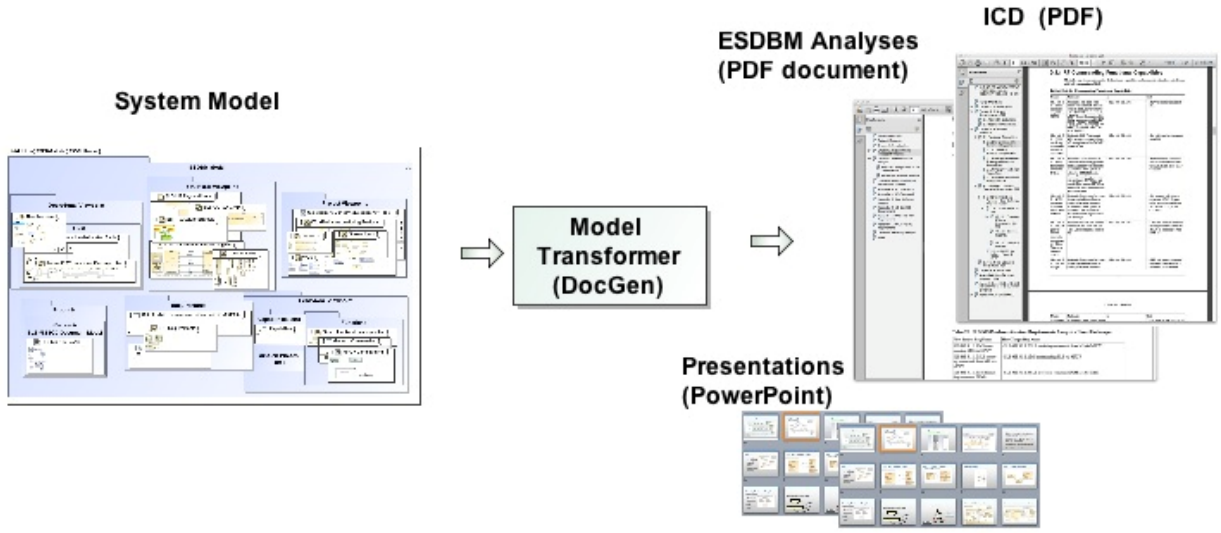


Figure 8.2-1 Automated Generation of Engineering Artifacts

The documents and other artifacts such as reports, power point presentations, etc., are produced from the system model using automated procedures that transform the system model into models of the artifacts.

One of the main benefits of MBSE emerges from incorporating all the information about the system into an *integrated* collection of interrelated models that represents the system from different perspectives (e.g., compositional, functional, operational, cost) with increased ability to correlate and retrieve any desired information. It ensures that data needed by programs and projects (e.g., for milestones, reviews, mission operations, and anomalies or investigations,

decisions, and outcomes) are identified and managed to provide traceability of the data used in decision-making. Interrelations defined between the model elements enhance the ability to maintain overall system representation consistency and enable efficient propagation of changes.

> *"Models have been used as part of document-based systems engineering approach for many years, and include functional flow diagrams, behavior diagrams, schematic block diagrams, N^2 charts, performance simulations, and reliability models, to name a few. However, the use of models has generally been limited in scope to support specific types of analysis or selected aspects of system design. The individual models have not been integrated into a coherent model of the overall system."* (Source: Friedenthal 2008)

8.2.2 MBSE Implementation

Employing MBSE on a particular program or project requires an underlying data and model foundation. The essential features of this foundation are (1) high-level system/architecture model(s); and (2) a capability to capture, manage, and access all system and programmatic data and their associated interrelationships.

There are multiple layers of models used throughout the broad range of engineering activities. These range from the lowest, most detailed, often discipline-specific models up to the generally descriptive, high-level architectural, functional, operational, and programmatic models. The high-level architectural models incorporate parameters from the system models and possibly the discipline-specific models to accurately represent the system. The level of abstraction decreases and the fidelity increases as the development of the system progresses through the life cycle. The table shown in Figure 8.2-2 illustrates and describes three layers of models that help illuminate the discussion of models as related to systems engineering.

1. The first layer – Model-Based Systems Engineering (MBSE) – often involves combining several activities from the systems engineering engine processes concurrently and iteratively, namely system behavior description, requirements analysis, system architecture, and test (V&V) approach. At this higher level, models may take the form of stand-alone or combined system behavior descriptions, requirements models, functional flow block diagram models, concept of operations models, programmatic work breakdown models, etc.
2. The second layer "bridges" Model-Based Systems Engineering (MBSE) and Model-Based Design (MBD). The system architecture defined in MBSE provides the organizing structure from which the discipline-specific models "hang." It may also be used for design space exploration and trade studies and includes simulation.
3. The third layer – Model-Based Design (MBD) – is usually used for detailed analysis and design, typically involving discipline-specific models and simulation software.

Lifecycle Activity *Levels of Abstraction*	Model Layer	Description
Concept & Architecture Definition *Functional/Logical Architecture*	1. Modeling and Specification (**MBSE**: Model-Based Systems Engineering)	A system is described with mostly **qualitative**, descriptive models, though, some models may be quantitative with first order or simple system relationships. These models can include system behaviors, high-level requirements, architectures, and functional and system structures. They are more generalized and, to some degree, may be executable.
Design Solution *System Architecture*	2. Modeling and First Simulation (**MBSE**: Model-Based Systems Engineering; **MBD**: Model-Based Design)	Models at this layer are mostly **quantitative** for the most part, incorporate multiple disciplines and can be simulated to measure performance against requirements (e.g., multi-physical simulation models); used for design space exploration and trade studies.
Detailed Design	3. Discipline-specific Modeling (**MBD**: Model-Based Design)	Models at this layer have a very **discipline-specific** character, e.g., geometry and CAE models; used for detailed analysis and design

Figure 8.2-2 Layers of Models Used throughout the Engineering Life Cycle

MBSE models support the various model-based engineering domains as illustrated in figure 8.2-3. Model layer 1 supports both the system engineering aspects of modeling and specification (i.e., MBSE) as well as use in model-based project control. Model layer 2 supports both MBSE and model-based manufacturing and operations. The design solution models enable the full manufacturing of the system as well as integrated operations of the system. Similarly, model layer 3 supports both model-based design and also specific manufacturing and operations of individual components, assemblies, and subsystems. Future experience will expand on the understanding and application of these model layers in the full Model-Based Engineering (MBE) domain.

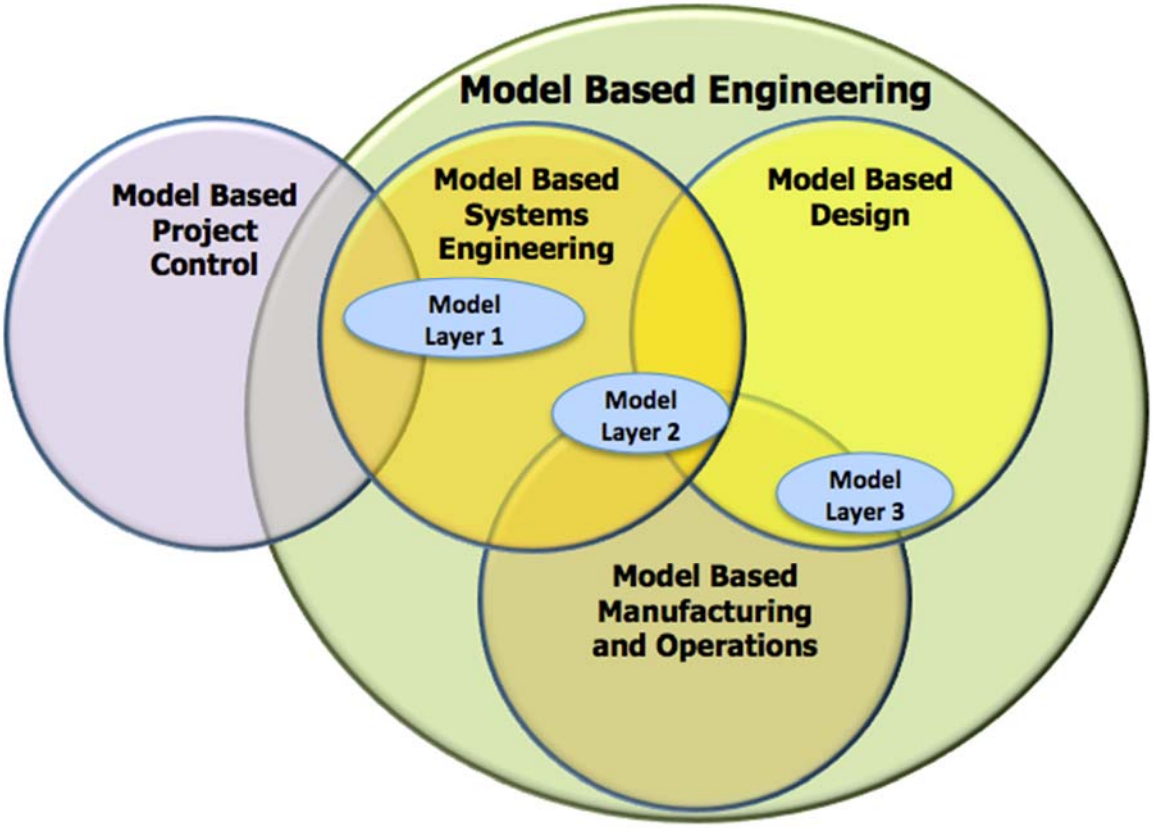


Figure 8.2-3 Notional View of Model-Based Engineering Relationships

The focus from an MBSE perspective is on these higher-level models. They may be in the form of stand-alone or combined requirements models, functional flow block diagram models, concept of operations models, programmatic work breakdown models, etc. A key feature would be the specification and capture of the interrelationships among data items within and between these models, such that one could, for example, perform a bi-directional trace between requirements, functions, and WBS products, whether or not they reside in the same or different models or databases.

The capability to capture, manage, and access data/interrelationships in models can be accomplished through a variety of methodologies, which range from the establishment of a single relational database to a virtually integrated, but distributed, database. The latter may be accomplished by means of a federation (or data map/index) of disparate data sources (see Figure 8.2-4). In all cases, the interrelationships (both within and between data sources) among the various data items are captured. Establishment of a "master map" or ontology (i.e., a common vocabulary for the types and attributes of the data items and their associated interrelationships) up front, for all these data items and their associated interrelationships, facilitates the establishment of this capability.

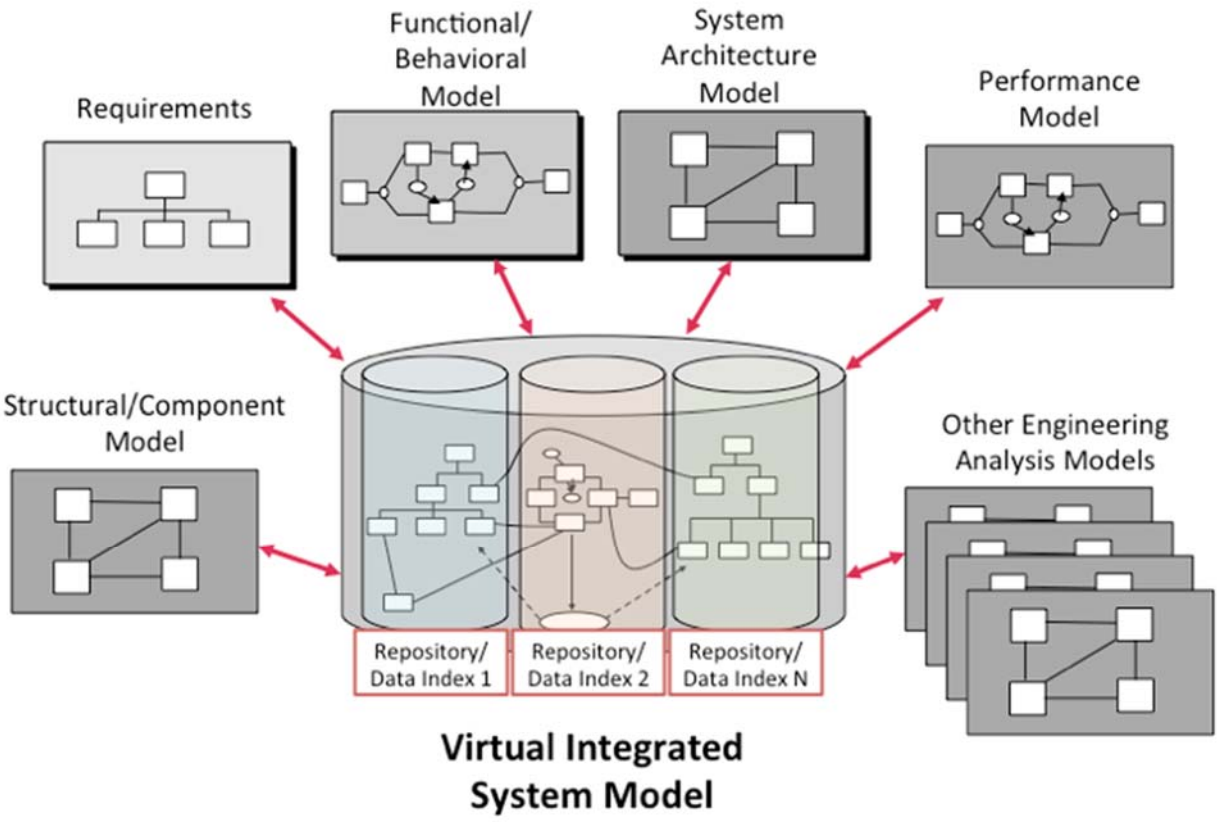


Figure 8.2-4 Virtually Integrated but Distributed Database

MBSE has the greatest benefit to a program, project, or activity when it is employed early in the lifecycle. New starts should consider MBSE approaches to improve risk posture and design efficiency. Applying MBSE later in a program, project, or activity requires a significant amount of rework to existing system models and documentation. Late in the design life cycle, many of the system integration decisions have been made, which greatly limits the utility of the MBSE application. Employing MBSE early allows the system engineer to make system integration decisions with a much clearer view of system integration issues.

8.2.3 The SE Engine and MBSE

This section illustrates how MBSE, as a crosscutting engineering initiative, may contribute to the SE engine processes (system design, product realization, technical management). Each section below examines how it is done traditionally, what limitations exist, and how MBSE helps.

8.2.3.1 System Design

Traditionally, the system design is captured using a variety of methods and is rendered in different forms from narratives and drawings to some partial models addressing particular aspects of the systems such as state charts or spreadsheets. A challenge with these descriptions is that they are difficult to integrate and their consistency is hard to prove, not to mention the traceability to the requirements definition data. Associations and relationships among the disparate data sources are provided in a model-centric approach that enables communication,

navigation, comparisons, version/configuration management, and aggregation of relevant information across the repositories. A model-centric approach allows the relevant information to be represented in a report rendered as a document, drawing, dynamic viewer, or whatever other form is suitable for the person (or machine) accessing the information.

A model-based approach helps address the problem of inconsistencies that may exist among various documents and spreadsheets and diagrams in a project. In the case of requirements, model elements representing requirements are related to model elements representing system components, functions, interfaces, and design analyses. If such a "system model" is built using well-defined types of elements and relationships (i.e., ontology) and following uniform process-driven modeling patterns, the resulting system model can be automatically analyzed for certain types of process-driven consistency and completeness.

With a model-based approach and modern standards and tools, the systems engineer represents the design in system descriptions and within that, defines the next lower level of the design (logical decomposition). In representing the design, the systems engineer can create models that provide a more coherent description of the system's design. Multiple views of the underlying model(s) may be created to enable the design (including many levels ranging from its high-level architectural principles to the detailed specifications at the component level) to be communicated and understood by the stakeholders and to enable them to verify that the system will address their concerns. A view may be arranged to present a selection of model elements (data, metadata, relationships, etc.), chosen to demonstrate that a particular set of concerns are indeed addressed by the design.

Also, a model-based approach enables the use of machine aids that enhance the ability to handle complexity; detect errors of completeness, consistency, and correctness. Throughout the design process, there is an effort to precisely and unambiguously express the analysis process, often formally defined in the system model, so that it can be determined why the analysis was necessary, what it analyzes, what the results are, etc.

Table 8.2-1 MBSE Contributions to System Design Processes

SE Engine System Design Processes	MBSE Contribution
Stakeholder Expectations Definition	Needs, goals, and objectives are kept within the model and form the top tier of eventual requirements flowdown. ConOps is modeled showing functional interconnections.
Technical Requirements Definition	Requirements are kept within the model allowing bi-directional traceability.
Logical Decomposition	Requirements can be categorized into functional, behavioral, performance, etc. These can be used to develop functional block diagrams, behavior diagrams, and other representations within the model.
Design Solution Definition	Allows integration of information and designs from different engineering domains providing consistency and traceability of the supporting analyses, and a single source of truth.

8.2.3.2 Product Realization

Product realization using traditional methods involves largely manual processes for creating the product and evaluating it based on the results of the system design processes. Many technical disciplines no longer practice manual product creation; for example, manufacturing and some kinds of software development are largely carried out using computer-aided processes. However, a document-centric systems engineering process must be translated into whatever model-centric forms are used for those computer-aided processes. This can be a labor-intensive step that is circumvented in MBSE, avoiding both the costs and the transfer errors that enter at this stage.

Integration of the end product in a traditional approach usually requires a hand-crafted integration plan. Also, deviations of the end product from the design may arise because of changes introduced during translation of a traditional expression of system designs into the model-based forms used for modern manufacturing. These deviations complicate the integration process, as do late-discovered errors in the implementation or higher-level system design.

With MBSE, integration constructs can be specified early in the design process, and can be exercised in early mission simulations to drive out system design errors. If the system design is model-based, conversion out of the traditional methods into model-centric manufacturing can be avoided, reducing the likelihood of implementation errors. As a result, the integration plan can be developed much more quickly, and if fully successful, "works the first time" integration can be achieved. The integrated model provides the necessary configuration status accounting to capture the as-designed, as-built, and as-delivered states of the product.

A model-based approach can maintain relationships between goals, requirements, designs, rationale, performances estimates, etc. up and down the systems engineering processes. This makes it easier to understand and communicate design changes, and to make corrections when needed.

Evaluation of products in a document-centric methodology can involve a laborious and time-consuming process of converting system specifications into verification plans and criteria. In an MBSE approach, using emerging tools, these are derived directly from the system specification in a largely automated fashion.

Table 8.2-2 MBSE Contributions to Product Realization Processes

SE Engine Product Realization Processes	MBSE Contribution
Product Implementation	Information captured in the system model can supplement additional product data to facilitate manufacturing.
Product Integration	Allows integrated system analysis to be conducted across all disciplines and supports system integration activities (i.e., hardware integration, software integration, hardware/software integration, human systems integration, and assembly integration).
Product Verification	Ability to tie test/analysis/demonstrations/inspections to specific requirements allows instant knowledge of verification progress.

SE Engine Product Realization Processes	MBSE Contribution
Product Validation	Ability to tie test/analysis/demonstrations/inspections to specific MOEs, expectations, and the ConOps scenarios allows instant knowledge of validation progress.
Product Transition	The product data package can be the fully integrated model as developed throughout the life cycle.

8.2.3.3 Technical Management

The crosscutting technical management functions of planning, control, assessment, and decision analysis continue to be needed in a model-based approach. The difference is that they are carried out using system models as the authoritative source of information.

For example, in a traditional approach, in contrast to a model-based approach, requirements are typically documented in a system requirements document first, and once the requirements are approved, they might be introduced in a database. The system architecture is often developed independently and somewhat in parallel. Once the high-level requirements and system architecture have been developed, systems engineers manually (or conceptually) trace the requirements to the architecture elements and adjust the architecture as needed. This process repeats at different levels, systems, subsystems, and components, accompanied by the tracing, again manually, of the derived requirements.

In a model-based approach, the requirements are expressed inside the system model first and traced to the evolving system design and other elements of the systems engineering process. The system model itself drives the configuration management processes, and when artifacts are needed for requirements and interface management, they are derived from the system model. This eases maintaining their mutual consistency by removing time-consuming obstacles to synchronizing and reconciling different management processes. In some cases, the models become the "requirement" rather than having derived requirement reports. For instance, a behavior requirement might be expressed as a sequence of activities in the model; having expressed that sequence once, there is no need to add other requirement statements. Verification can be performed using the model as the reference to which the actual system is compared during evaluation activities.

Table 8.2-3 MBSE Contributions to Technical Management Processes

SE Engine Technical Management Processes	MBSE Contribution
Technical Planning	Systems engineering processes can be modeled and the elements of the model can be related to the WBS and the project master plans and schedules providing insights for better planning and replanning.
Requirements Management	With the enhanced ability to trace requirements to their source and implementation, proposed changes can be modeled to determine cost, schedule, and/or technical impacts on the product.

SE Engine Technical Management Processes	MBSE Contribution
Interface Management	Interfaces captured in the model can be automatically checked for compatibility, changes can be made and their impacts identified.
Technical Risk Management	Models can be used to identify potential risks and to help determine their cost, schedule, and technical impacts.
Configuration Management	By identifying and managing a single source baseline within a model-based system, configuration management processes are optimized and simplified.
Technical Data Management	By identifying, characterizing, and controlling technical data, metadata, and the exchanges of data within the model, data can be better visualized, optimized, and distributed to the data actors.
Technical Assessment	Models can be used to present and visualize the information at life-cycle reviews as well as provide a means for reviewers to check for flaws.
Decision Analysis	Trade studies can be performed quickly by varying parameters within the model to determine their impact on the overall design, providing key information to the decision-makers.

8.2.4 Models

As its name implies, model-based systems engineering relies on the creation and use of a set of identified models which individually capture specific portions and/or views of the key computational and descriptive aspects of a specific mission or system. Models should be developed for a purpose. They should address specific stakeholder concerns/needs and have a clear usefulness to engineering the system. In a model-centric environment, the data is captured once and represented many times based on the defined viewpoint of the system description. The total set of integrated models represents the complete system as it exists or will exist at some point in time.

Along with other commonly-used structures such as the Work Breakdown Structures (WBSs) and Product Breakdown Structures (PBSs) called for in NPRs 7120.5 and 7123.1, identification of a set of unambiguous and commonly-structured mission and system reference models that are utilized and maintained throughout the life cycle will provide further structure and focus to the engineering processes, enhancing the ability to visualize change and manage system complexity, and improve the quality of engineering products earlier in the mission life cycle.

When the question of reusing a system or architecture model arises, the systems engineer should pay close attention to the interconnections and ontology used in the original model. It may be difficult to identify the necessary changes to the model in the new system context and care should be taken to ensure this can be accomplished. In software, the rule of thumb is to build new software when more than ~20% of the code must be rewritten. It is anticipated that models will have a similar threshold and more data needs to be taken to establish this for modeling.

8.2.4.1 Modeling Languages

Whether a model is a "computational," math-based model or only a descriptive one, a graphical modeling language may be identified and used to capture and represent the contents and relationships depicted in each of the models. Using the modeling language's formal composition structure or pattern (syntax), its defined vocabulary and rules (semantics), the objects and the relationships depicted in each of the models will be documented and aligned in a structured format that may be precisely understood by a computer and utilized to represent the processes and data graphically.

There is no universal language that can cover all conceivable systems, so a reasonable solution is to define domain-specific modeling languages. Even a domain-specific language can become too general if more specialized subdomains are involved. It is reasonable to conceive a family of modeling languages having a common conceptual core and branching out to more specialized concepts. Currently, the Systems Modeling Language (SysML) is an often used modeling language for systems design.

Given the imprecise nature of SysML semantics, even if the language can be extended to better represent the domain where it is applied, there is still no guarantee that the resulting models are semantically accurate. Using a modeling language that has formal semantics that can be automatically verified may alleviate this problem. An example of such a language is the Web Ontology Language (OWL). Another possible solution is to use the Object Constraint Language (OCL) in conjunction with SysML, a "within SysML" approach.

8.2.4.2 Model-Based Vocabularies

To communicate a common understanding of the required content and context for using a specific mission or system model, a formal description of the data and required content within the model must be developed and shared across participating organizations. Typically, this representation takes the form of a formal specification (ontology) or common vocabulary that identifies the kinds of conceptual objects (classes or sets of things), attributes (properties), and the relationships that may exist among these objects within a specific domain or area of interest. (See an example in Figure 8.2-5.)

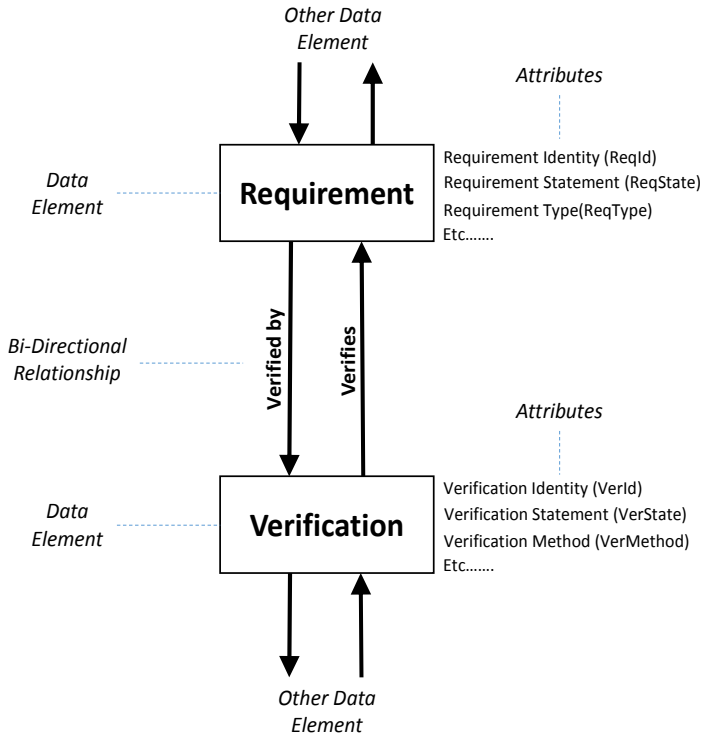


Figure 8.2-5 Data Ontology Example (Requirement and Verification)

This vocabulary serves as the foundation for organizing information across the models and the domain and ensures cohesion and commonality by representing the things, ideas, events and their properties and relations according to specific categories. Additional information and rules related to the modeled objects' meaning and the relationships that must be maintained are provided to support interoperability across disciplines and disparate systems. In some cases, portions of this information may be commonly documented in an organization's naming and data identification conventions, a model catalog, business reference architecture, or other similar formats.

8.2.4.3 Modeling Standards

Modeling standards play an important role in supporting MBSE's goals by promoting increased understandability, communication, and integration of models. There are various areas of modeling that may be subject to standardization, including modeling languages, the transfer of information from one model to another, and model transformations. A well-defined ontology and methodology (or methodologies), applied consistently, are extremely important.

Modeling language standards include languages defined for more traditional systems engineering approaches such as the Functional Flow Block Diagram (FFBD) (see Oliver 1997) and the Integration Definition for functional modeling (IDEF0) (see Knowledge Based Systems, Inc. (KBSI)), as well as more recent generic systems engineering standards (brought by MBSE) such as SysML and OWL (see language-specific references).

Systems architecture is the object of several architecture standards such as the Unified Profile (UPDM) for the (U.S.) Department of Defense Architecture Framework (DoDAF) and the (U.K.)

Ministry Of Defence Architecture Framework (MODAF), and in particular, the ISO/IEC/IEEE 42010:2011 standard, *Systems and Software Engineering — Architecture Description*.

Examples of standards for enabling model interoperability through the transfer of information between models include the Application Protocol for Systems Engineering Data Exchange (AP-233) and the Extensible Markup Language (XML) Metadata Interchange (XMI).

Model interoperability at the conceptual level and model transformation are enabled by generic standards such as Query View Transformations (QVT) or by specific transformation standards such as the Systems Modeling Language (SysML)-Modelica Transformation. (See Paredis 2010.)

There are also standards developed for specific domains such as software design (Architecture Analysis and Design Language (AADL) (see SAE 2012)), hardware design (Very-High-Speed Integrated Circuit (VHSIC) Hardware Description Language (VHDL)), and business processes (Business Process Modeling Notation (BPMN).

8.2.5 MBSE Methodologies

Traditionally, systems engineering has been performed using a document-centric methodology, using paper or electronic document. A MBSE methodology merges the best practices of systems engineering with the use of modeling. Transitioning an organization from a document-based systems engineering methodology to a model/data-centric methodology does not alter the underlying, well-understood processes that are already in place.

A process-based methodology provides an understanding of what information is needed to effectively execute the program/project's processes and provides a framework for effectively managing its information environment. It provides identification, management, interoperability, and integration of information across programmatic and technical domains needed to support Product Data Life-cycle Management (PDLM) goals.

NPR 7120.9, Product Data and Life-Cycle Management (PDLM) for Flight Programs and Projects, and the associated *NASA-HDBK-0008*, *NASA PDLM Handbook,* describe the responsibilities and requirements for effectively managing authoritative data that defines, describes, analyzes, and characterizes a product throughout its life cycle. It includes requirements for establishing four types of architectures: security architecture, information support system architecture, process architecture, and data architecture. The first two can be viewed as MBSE "*enabling*" architectures and are considered part of the underlying IT infrastructure. The focus of the program/project is in establishing the process and data architectures.

The foundation to any MBSE development is a well-defined and understood set of process and data architectures. These architectures provide the guidelines and road maps for implementing the model-based approach. The process architecture is already well established and has been used via a Document-Based Systems Engineering (DBSE) methodology. MBSE does not change the underlying defined and approved processes. What is changing, however, is the methodology for implementing these processes.

Similar to the process architecture definition, many elements and viewpoints of the data architecture may have already been defined in NPRs from a document standpoint. These data architecture elements are easily extractable from document-based forms, templates, Excel spreadsheets, etc. The main effort in this step is to extract the data objects, attributes, and association to other data objects contained within the document-based forms and templates. The resulting "ontology" is an agnostic tool and represents the standard relationship shown in a document-centric environment. For example, verifications verify requirements, which specify an architecture used to achieve a mission that is guided by needs, goals and objectives. This is an example of an entity-relational model between NGOs, Design Reference Missions (DRMs), products/architecture, and requirements. It is important to keep in mind that both document-centric and model-centric systems engineering share a common process. The difference is in the methodology (or methodologies) for implementing those processes.

The next step is to ensure that the data architecture artifacts map to and meet the intent of the process architecture. In some cases, document-based systems can mask process deficiencies due to the vast amounts of disparate data spread across multiple documents. Model-Based Systems Engineering (MBSE) tools need to be able to "expose" data to other applications for data integration and interrogation. These tools are typically related to a specific systems engineering or program management function; i.e., requirements management, risk management, schedule management, budget management, etc. There are several broad spectrum applications on the market that can satisfy multiple functions, making data integration a less burdensome task. For smaller programs/projects, this approach is preferred to limit the overhead of developing complex information technology systems to integrate the disparate data.

NPR 7123.1 establishes "the core set of common technical processes and requirements to be used by NASA projects in engineering system products during all life-cycle phases to meet phase exit criteria and project objectives" as the approved systems engineering processes at NASA. MBSE methodologies do not change the overarching SE processes expressed in NPR 7123.1; rather, MBSE provides a more effective way of carrying out parts, or in some cases, all of the process. In fact, the NASA SE processes expect that a model-based approach will be used where appropriate, as evidenced by statements in 7123.1 such as the following:

> *"...technical teams and individuals should use the appropriate and available sets of tools and methods to accomplish required common technical process activities. This would include the use of modeling and simulation as applicable to the product-line phase, location of the WBS model in the system structure, and the applicable phase exit criteria." (Source: Section 3.1.2.5)*

Models explicitly mentioned within NPR 7123.1 as being part of the NASA SE processes include logical decomposition models, functional flow block diagrams, timelines, data control flow, states and modes, behavior diagrams, operator tasks, and functional failure modes, although these are certainly not all the possible applications of MBSE.

Note that document-based systems engineering is a methodology that uses documents to accomplish the NASA SE processes. The document templates provided in the NPRs and handbooks can be interpreted as being view specifications that explain how to represent "models" populated with program/project data. The difference is that in document-based systems

engineering, the views are produced in an undefined and potentially inconsistent or incomplete manner from the program/project data residing in disparate models scattered among many spreadsheets, tools, and individual minds. In MBSE, the program/project data resides in an underlying integrated system model(s), representing a single source of truth and possessing well-defined rules for generating the required artifacts from the model. MBSE also differs from document-based systems engineering in that the digital form often used for the integrated system model provides an opportunity for machine access to handle jobs that are intractable or excessively laborious when carried out by hand, such as error checking, change propagation, model transformations, and artifact generation.

8.2.6 MBSE Implementation Challenges

Partial employment of MBSE methodologies can enable many of the benefits noted above, but for the full value of MBSE to be realized, broader enabling and supporting capabilities need to be established. There are several challenges that must be addressed for full implementation of MBSE that range from the enabling IT infrastructure to the development of system models and ontologies to organizational and cultural change.

8.2.6.1 Establishment of IT Infrastructure

NPR 7120.9 defines requirements for establishing the enabling security and information support system architectures. As these IT infrastructures are further developed and deployed, a more comprehensive MBSE culture is enabled. In the interim, programs need to tailor their MBSE methodologies to "fit" existing IT capabilities and/or provide their own gap-filling capabilities. In a similar manner, programs need to provide access to the necessary MBSE-related tools, applications, and aids that are not yet institutionally-provided.

8.2.6.2 User Interface Usability

An important attribute of an effective MBSE capability is the ability of the user (e.g., engineer, analyst, decision-maker, etc.) to search and gain access to data/information that meets the user's needs in a form that is easily comprehensible. According to NPR 7120.9: "Facilitate the provision of comprehensive search and integrated views (including reports) of data with a high degree of usability." Ready access to easily understandable, comprehensive data and information may always be a challenge, but proper attention to the underlying data architecture and to interoperability enables development of improved user interface capabilities.

8.2.6.3 Establishment of Ontology

A critical element for enabling MBSE is the establishment of an ontology. Different disciplines tend to call the same components, effects, or events by different names. This creates confusion during integration. Defining a program-specific ontology (defining the data types, attributes, and interrelationships) is important to help eliminate this confusion. This may require significant effort and is best done early in the program. Early establishment of an ontology provides benefits as the program progresses and becomes more detailed and complex. Over time, similar programs and common processes/functions will converge on common core portions of ontologies that can be employed in new starts, thus lowering the barriers to adopting MBSE.

8.2.6.4 Development of High-Level System Model(s) and Associated Database(s)

Modeling is a specialized skill and may require a dedicated modeler at the system level fed by inputs from Subject Matter Experts (SME) in each of the system functions or disciplines. This allows the full syntax of the system modeling tool to be employed and can improve the integration of the model. A skilled system and architecture modeler can also aid in modeling constructs that are more intuitive and that readily visualize key system characteristics.

8.2.6.5 Configuration Management

Configuration management presents new challenges in the Model-Based Engineering (MBE) environment. Using traditional configuration management practices, collections of documentation comprise the configuration information that defines any particular Configuration Item (CI). The MBE environment changes the range of configuration information developed to include performance and design models, database objects, as well as more traditional book-form objects and formats. Traceability between these objects must also be identified and controlled using available capabilities. Additionally, users will need various views and enabling capabilities such as snapshots of baselines, version releases and freezes, and status and account metrics and reports for each identified CI.

8.2.6.6 Contractual Practices and Technical Data Management

Data requirements have traditionally been used as the primary data management process for technical management of documents and drawings between NASA, their development contractors, and in-house design activities. A simple transition to using the same process for technical data management of electronic documents and drawings (e.g., Computer-Aided Design (CAD) models) does not fully take advantage of the benefits of MBE. Technically managed bidirectional exchange or access to the full range of levels of models and databases is a capability that has not yet been realized. Refinements to contractual practices and technical data management processes, including the underlying objectives and associated processes as well as the enabling contractual language and technical capabilities, must be developed to support a model-based environment.

8.2.6.7 Organizational and Cultural Challenges

A common challenge to implementation of any new paradigm, process, or capability is managing organizational and cultural change. Recognition of the often convincing reasons and methods to "stay the course" is important to developing effective and compelling approaches to managing these changes. Education, training, and access to the necessary tools, applications, and aids can be helpful in this regard. In general, lowering barriers to adoption and implementation is necessary.

8.2.7 MBSE Benefits

Model-based systems engineering does not affect process but will enable the opportunity for overall better quality, lower cost, and lower risk for several reasons. These benefits come about because:

- There can be greater consistency of all products because any single piece of design information can be expressed authoritatively in a single place that can later be referred to by others for decisions, derivations, or formation of artifacts.
- There can be better visibility into the salient characteristics of a system because multiple views can be created that succinctly address specific stakeholder concerns.
- There can be greater congruence between documentation and reality:
  - Model-based artifacts can be generated automatically, lowering the effort to keep them up to date with the result that artifacts can always match the best available information.
- Navigation, traceability, and interrogation of information are facilitated in the model-based approach. People can have access to the information they are authorized to have more quickly and on an as-needed basis without going through manual distribution or search processes.
- Models used for verification can have higher quality, and provide greater confidence if design and manufacturing models are applied diligently before and after use of the verification models.
- Models themselves can help to reveal hidden flaws of the models.
- There can be less investment lost in erroneous design because sometimes the model reveals a flaw as soon as it is created, enabling correction before downstream work is done, work that would be invalid if the upstream mistake were not corrected immediately.
- Having fewer inconsistencies between artifacts lowers the costs for verification.
- It provides identification, management, interoperability, and integration of information across business or organizational elements needed to support program PDLM goals.
- It ensures that data needed by programs and projects (e.g., for milestones, reviews, mission operations, and anomalies or investigations, decisions, and outcomes) are identified and managed to provide traceability of the data used in decision-making.

8.3 Concept Maturity Levels

Concept Maturity Levels[5] (CMLs) were introduced to provide mission architects and systems engineers with a way to measure and communicate the fidelity and accuracy of a mission concept during the early stages of its life cycle. The CMLs represent a scale that provides a repeatable way to assess and describe the maturity of concepts and a single numerical scale, comparable to the TRL scale, to assess the maturity of different mission concepts.

Mission concept development teams use this method and associated tools throughout the pre-project study phase and on through the Formulation phases (Phase A/B). Prior to the advent of the CML scale, there were no standardized methods available to (1) determine how much work was placed into a mission concept; (2) explicitly know when in a pre-project's life-cycle trade space exploration would be most advantageous to ensuring that a mission concept was the most scientifically relevant and cost-effective; (3) determine which concepts had the same level of work and could be compared on the same terms; and (4) how much work a mission concept required to achieve a subsequent level of maturity.

The CML organizing structure corresponds to an increasing level of maturity as the concept, design, implementation, and risks are analyzed and evolve. The key strength of CMLs is the ability to measure mission concept maturity guided by an incremental set of maturity characteristics that is separately developed and corresponds to the particular type of mission; e.g., robotic, human, airborne, ground-based missions. (See Wessen 2013.)

The fidelity of a concept is how closely it resembles an idealized system, while accuracy is the correctness of the estimate within a certain threshold. The CML vocabulary provides a standardized mechanism for describing and communicating the products/accomplishments required for achieving a given CML and for identifying the work remaining before proceeding to the next level. CMLs address the broad scope of engineering, science, and programmatic parameters, and are useful for identifying analysis gaps and areas requiring more in-depth evaluation. It is important to note that for each CML level achieved, the system fidelity, system accuracy, and their implementation are more clearly understood. Consequently, the risk posture of the system is generally lowered and "work to go" should be better understood.

Figure 8.3-1 shows the CMLs across the concept development and formulation phases.

[5] The idea for Concept Maturity Levels was developed in 2008 by the JPL Strategic Planning & Project Formulation Office's Chief Engineer, Dr. Mark Adler, and reflects his concept for identification of evolving mission maturity and its assessment. This section relies heavily on these concepts and the reference paper and the contributions of Randii Wessen and Jairus Hihn from the Jet Propulsion Laboratory, Pasadena, CA.

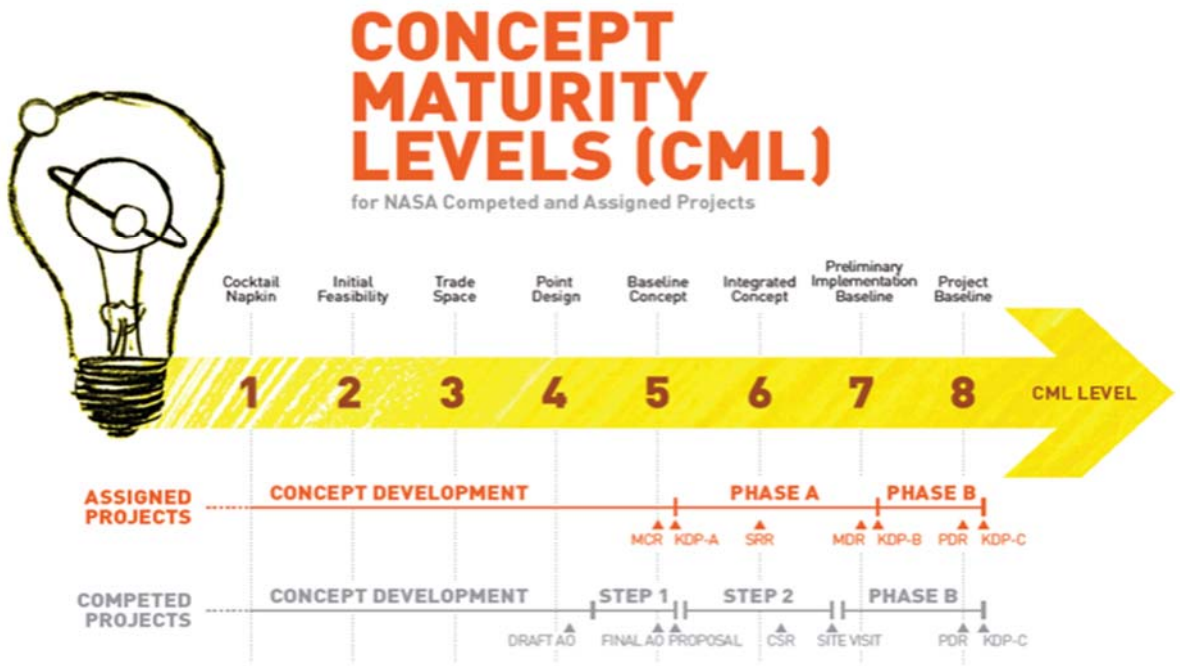


Figure 8.3-1 CML for NASA Competed and Assigned Projects

A description of each CML is provided in Table 8.3-1.

Table 8.3-1 Description of Concept Maturity Levels

CML	Name	Description
1	*Cocktail Napkin*	The science questions have been well articulated, the type of science observations needed for addressing these questions have been proposed, and a rudimentary sketch of the mission concept and high-level objectives have been created. The essence of what makes the idea unique and meaningful has been captured.
2	*Initial Feasibility*	The idea is expanded and questioned on the basis of feasibility from a science, technical, and programmatic viewpoint. There is basic understanding of the science and mission needs and the concepts for achieving these. Lower-level objectives have been specified, key performance parameters quantified, and basic calculations have been performed. These calculations, *to first order*, determine the viability of the concept.
3	*Trade Space*	Exploration has been done around the science objectives and architectural trades between the spacecraft system, ground system, and mission design to explore impacts on and understand the relationship between science return, cost, and risk. Typically, this results in identified risks that will need to be investigated and possible mitigation.
4	*Point Design*	A specific design and cost that returns the desired science has been selected within the trade space and defined down to the level of major subsystems with acceptable margins and reserves. Subsystems trades have been performed.

CML	Name	Description
5	*Baseline Concept*	Implementation approach has been defined including partners, contracting mode, integration and test approach, cost and schedule. This maturity level represents the level needed to write a NASA Step 1 proposal (for competed projects) or hold a Mission Concept Review (for assigned projects).
6	*Integrated Concept*	Expanded details on the technical, management, cost, and other elements of the mission concept have been defined and documented. A NASA Step 2 Concept Study Report (CSR) is at this level of maturity. A corresponding milestone for assigned projects is the System Requirements Review (SRR).
7	*Preliminary Implementation Baseline*	Preliminary system- and subsystem-level requirements and analyses, demonstrated (and acceptable) margins and reserves, prototyping and technology demonstrations, risk assessments and mitigation plans have been completed.
8	*PDR (Project Baseline)*	Design and planning commensurate for a Preliminary Design Review (PDR) driven by NPR 7120.5 gate products during the project implementation.
9	*CDR*	Design and planning commensurate for a Critical Design Review (CDR) driven by NPR 7120.5 gate products during the project implementation.

The CML concept has proved very useful when comparing the products of different CE or design teams. (See Chattopadhyay.) A CE or design team is typically formed for the purpose of developing products for a particular CML range as a CE team needs very different infrastructure, tools, and processes to deal with each CML. It is challenging to make a single team-type fit all concept levels. The type of products the team generates – whether it is a higher-level architecture comparison or an elaborate point design – is also a result of this choice of CML range.

Architecture teams such as JPL's A-Team and Goddard's Architecture Laboratory primarily work at CML 2 with a range of CML 1-3. CML 1 and CML 2 design teams work with open-ended ideas, often considering a wide variety of science objectives and mission architectures. The intent of the CML 1-3 process is to create innovative missions that respond to the science and programmatic (e.g., cost cap) needs and evaluate them to a sufficient level of detail so that the most promising mission concepts are identified. The estimates of key parameters such as mass, power, and cost have relatively large uncertainty ranges, even as high as +/- 50% or larger.

CML 4 teams such as Goddard's Mission Design Lab (MDL) team and JPL's Team X primarily deal with point designs and primarily work at CML 4 but can range from CML 3-5. In contrast to lower-level teams, CML 4 teams analyze variations around a specified point design. CML 4 teams start with what is equivalent to the output of a CML 3 team, so a specific mission has been identified and many of the subsystem elements have been identified. The tools used are much higher fidelity and accuracy with estimates in +/- 25%.

It becomes clear by looking at design teams this way that a CML 2 team can never work concurrently as an extension of a CML 4 team. Different CML teams need to work serially as the output of a lower-level team is the input to higher-level CML teams.

Finally, CMLs have been expanded into a concept maturity level matrix. This matrix is used to determine the CML level for a specific mission concept. (See Wessen 2013.)

Appendix A: Acronyms

AADL	Architecture Analysis and Design Language
ABC	Agency Baseline Commitment
ACWP	Actual Cost of Work Performed
AD^2	Advancement Degree of Difficulty Assessment
AHP	Analytic Hierarchy Process
AIAA	American Institute of Aeronautics and Astronautics
AO	Announcement of Opportunity
AoA	Analysis (or Analyses) of Alternatives
AS9100	Aerospace Quality Management Standard
ASME	American Society of Mechanical Engineers
ASQ	American Society for Quality
ASRB	Airworthiness Safety Review Board
ATD	Advanced Technology Development
BAC	Budget at Completion
BAR	Basic and Applied Research
BCWP	Budgeted Cost for Work Performed
BCWS	Budgeted Cost for Work Scheduled
BPMN	Business Process Modeling Notation
CAD	Computer-Aided Design
CAE	Computer-Aided Engineering
CAIB	Columbia Accident Investigation Board
CAM	Control Account Manager or Cost Account Manager
CATEX	Categorical l Exclusion (NEPA)
CBE	Current Best Estimate
CCB	Configuration Control Board
CDR	Critical Design Review
CE	Concurrent Engineering or Chief Engineer
CEQ	Council on Environmental Quality
CERR	Critical Event Readiness Review
CHSIP	Commercial Human Systems Integration Processes
CI	Configuration Item
CM	Configuration Management
CMC	Center Management Council
CML	Concept Maturity Level
CMO	Configuration Management Organization
CNM	(NASA) Center NEPA Manager
CNSI	Classified National Security Information
ConOps	Concept of Operations
COSPAR	Committee on Space Research
COTS	Commercial Off-The-Shelf
CP	Commercial Partner
CPI	Critical Program Information or Cost Performance Index

CR	Change Request
CRM	Continuous Risk Management
CSA	Configuration Status Accounting
CSR	Concept Study Report
CWBS	Contract Work Breakdown Structure
ΔV	Delta-Velocity
D&C	Design and Construction
DBSE	Document-Based Systems Engineering
DCR	Design Certification Review
DDT&E	Design, Development, Test, and Evaluation
DM	Data Management
DMS	Diminishing Manufacturing Sources
DOD	(U.S.) Department of Defense
DODAF	DOD Architecture Framework
DOE	(U.S.) Department of Energy
DR	Decommissioning Review
DRC	Design Reference Case
DRM	Design Reference Mission
DRR	Disposal Readiness Review
EA	Environmental Assessment
EAC	Estimate at Completion
ECLSS	Environmental Control and Life Support Systems
ECP	Engineering Change Proposal
ECR	Engineering Change Request
EDL	Entry, Descent, and Landing
EDU	Engineering Development Unit
EEE	Electrical, Electronic, and Electromechanical
EFFBD	Enhanced Functional Flow Block Diagram
EIA	Electronic Industries Alliance
EIS	Environmental Impact Statement
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
EMO	Environmental Management Office
EO	(U.S.) Executive Order
EOM	End of Mission
EPA	(U.S.) Environmental Protection Agency
EPS	Electrical Power System
ESTEC	European Space Research and Technology Center
ET	External Tank
EV	Earned Value
EVM	Earned Value Management
FA	Formulation Agreement
FAD	Formulation Authorization Document

FAR	Federal Acquisition Regulation
FCA	Functional Configuration Audit
FDIR	Failure Detection, Isolation, And Recovery
FE	Flight Element
FFBD	Functional Flow Block Diagram
FIPS	Federal Information Processing Standard
FM	Fault Management
FMEA	Failure Modes and Effects Analysis
FMECA	Failure Modes, Effects, and Criticality Analysis
FMR	Financial Management Requirements
FMSE	Fault Management Systems Engineer
FONSI	Finding Of No Significant Impact
FRR	Flight Readiness Review
FSAR	Final Safety Analysis Report (DOE)
FTE	Full Time Equivalent
GEO	Geostationary
GFP	Government-Furnished Property
GMIP	Government Mandatory Inspection Point
GOTS	Government Off-The-Shelf
GPS	Global Positioning Satellite
GRC	Goddard Research Center
GSE	Government-Supplied Equipment or Ground Support Equipment
GSFC	Goddard Space Flight Center
HCD	Human-Centered Design
HF	Human Factors
HFE	Human Factors Engineering
HITL	Human In The Loop
HQ	Headquarters
HQ/EMD	(NASA) Headquarters/Environmental Management Division
HSI	Human Systems Integration
HSIP	Human System Integration Plan
HWIL	HardWare In the Loop
I&T	Integration and Test
I&V	Integration and Verification
ICD	Interface Control Document/Drawing
ICP	Interface Control Plan
IDD	Interface Definition Document
IDEF0	Integration Definition (for functional modeling)
IEEE	Institute of Electrical and Electronics Engineers
ILS	Integrated Logistics Support
INCOSE	International Council on Systems Engineering
INSRP	Interagency Nuclear Safety Review Panel
IP	International Partner

IPEP	IV&V Project Execution Plan
IPT	Integrated Product Team
IRD	Interface Requirements Document
IRN	Interface Revision Notice
ISO	International Organization for Standardization
Isp	Specific Impulse
IT	Information Technology or Iteration
ITA	Internal Task Agreement
ITAR	International Traffic in Arms Regulation
IV&V	Independent Verification and Validation
IVHM	Integrated Vehicle Health Management
IWG	Interface Working Group
JCL	Joint (cost and schedule) Confidence Level
JPL	Jet Propulsion Laboratory
KBSI	Knowledge Based Systems, Inc.
KDP	Key Decision Point
KDR	Key Driving Requirement
KPP	Key Performance Parameter
KSC	Kennedy Space Center
KSI	Kilopounds per Square Inch
LCC	Life-Cycle Cost
LCCE	Life-Cycle Cost Estimate
LEO	Low Earth Orbit or Low Earth Orbiting
LLIS	Lessons Learned Information System
LOC	Loss Of Crew
LOM	Loss Of Mission
LP	Launch Package
LSE	Lead Systems Engineer
M&S	Modeling and Simulation or Models and Simulations
MAUT	Multi-Attribute Utility Theory
MBD	Model-Based Design
MBE	Model-Based Engineering
MBSE	Model-Based Systems Engineering
MCDA	Multi-Criteria Decision Analysis
MCR	Mission Concept Review
MDAA	Mission Directorate Associate Administrator
MDR	Mission Definition Review
MEL	Master Equipment List
MFR	Memorandum For Record (NEPA)
MGA	Mass Growth Allowance
ML/MP	Multi-Level, Multi-Phase
MODAF	(U.K.) Ministry of Defence Architecture Framework

MOE	Measure of Effectiveness
MOP	Measure of Performance
MOTS	Modified Off-The-Shelf
MOU	Memorandum of Understanding
MRB	Material Review Board
MRR	Mission Readiness Review
MSE	Mission Systems Engineer
MSFC	Marshall Space Flight Center
N2	N-squared (diagrams)
NASA	(U.S.) National Aeronautics and Space Administration
NASA-TLX	NASA Task Load Index
NEDT	NASA Exploration Design Team not used
NEN	NASA Engineering Network
NEPA	National Environmental Policy Act
NETS	NASA Environmental Tracking System
NFS	NASA FAR Supplement
NGO	Needs, Goals, and Objectives
NIAT	NASA Integrated Action Team
NID	NASA Interim Directive
NLSA	Nuclear Launch Safety Approval
NOA	New Obligation Authority
NOAA	(U.S.) National Oceanic and Atmospheric Administration
NODIS	NASA Online Directives Information System
NPD	NASA Policy Directive
NPR	NASA Procedural Requirements
NRC	(U.S.) Nuclear Regulatory Commission
NSES	NASA Statistical Engineering Symposium
NSTS	National Space Transportation System
OCE	(NASA) Office of the Chief Engineer
OCHMO	(NASA) Office of the Chief Health and Medical Officer
OCIO	(NASA) Office of the Chief Information Officer
OCL	Object Constraint Language
OCSO	Organizational Computer Security Officer
OIIR	(NASA) Office of International and Intergovernmental Relations
OMB	(U.S.) Office of Management and Budget
OMG	Object Management Group, Inc.
ORR	Operational Readiness Review
OSTP	(U. S.) Office of Science and Technology Policy
OTS	Off-the-Shelf
OWL	Web Ontology Language
PA	Product Assurance
PBS	Product Breakdown Structure
PCA	Physical Configuration Audit or Program Commitment Agreement

PDLM	Product Data Life-cycle Management
PD/NSC	(U.S.) Presidential Directive/National Security Council
PDR	Preliminary Design Review
PERT	Program Evaluation and Review Technique
PFAR	Post-Flight Assessment Review
PHA	Preliminary Hazard Analysis
PI	Performance Index or Principal Investigator
PIR	Program Implementation Review
PIRN	Preliminary Interface Revision Notice
PKI	Public Key Infrastructure
PLAR	Post-Launch Assessment Review
P(LOC)	Probability of Loss of Crew
P(LOM)	Probability of Loss of Mission
PM	Program Manager or Project Manager
PMB	Performance Measurement Baseline (EVM)
PMC	Program Management Council
PPBE	Planning, Programming, Budgeting, and Execution
PPD	(U.S.) Presidential Policy Directive
PPO	Planetary Protection Officer
PPP	Program/Project Protection Plan
PQASP	Program/Project Quality Assurance Surveillance Plan
PRA	Probabilistic Risk Assessment
PRD	Project Requirements Document
PRR	Production Readiness Review
PSAR	Preliminary Safety Analysis Report (DOE)
PSI	Pounds per Square Inch
PSR	Program Status Review
QA	Quality Assurance
QVT	Query View Transformations
R&D	Research and Development
R&M	Reliability and Maintainability
R&T	Research and Technology
RACI	Responsible, Accountable, Consulted, Informed
REC	Record of Environmental Consideration
RF	Radio Frequency
RFA	Requests for Action
RFI	Request for Information
RFP	Request for Proposal
RHU	Radioisotope Heater Unit
RID	Review Item Discrepancy or Review Item Disposition
RIDM	Risk-Informed Decision-Making
ROD	Record of Decision
ROM	Rough Order of Magnitude
RM	Risk Management

RMA	Rapid Mission Architecture
RPS	Radioisotope Power System
RTE	Responsible Test Engineer
RUL	Remaining Useful Life
SAR	System Acceptance Review or Safety Analysis Report (DOE)
SBU	Sensitive But Unclassified
SDR	Program / System Definition Review
SE&I	Systems Engineering and Integration
SE	Systems Engineering
SECoP	Systems Engineering Community of Practice
SEE	Single-Event Effect
SEMP	Systems Engineering Management Plan
SER	Safety Evaluation Report
SI	International System of Units (French: Système international d'unités)
SIR	System Integration Review
SL/SP	Single Level, Single Phase
SMA	Safety and Mission Assurance
SME	Subject Matter Expert
SMSR	Safety and Mission Success Review
SOW	Statement Of Work
SP	Special Publication
SPI	Schedule Performance Index
SRB	Standing Review Board or Solid Rocket Booster
SRD	System Requirements Document
SRR	Program / System Requirements Review
SRS	Software Requirements Specification
SSA	Space Situational Awareness
STI	Scientific and Technical Information
STS	Space Transportation System
SysML	System Modeling Language
T&E	Test and Evaluation
TA	Technology Assessment
TBA	To Be Announced
TBD	To Be Determined
TBR	To Be Resolved
TD	Technology Development
TDRSS	Tracking and Data Relay Satellite System
TLA	Timeline Analysis
TLS	Timeline Sheet
TMA	Technology Maturity Assessment
TOC	Turn Over Cart
ToR	Terms of Reference
TPM	Technical Performance Measure
TPS	Thermal Protection System

TQM	Total Quality Management
TRA	Technology Readiness Assessment
TRAR	Technology Readiness Assessment Report
TRL	Technology Readiness Level
TRR	Test Readiness Review
TVC	Thrust Vector Controller
UFE	Unallocated Future Expenses
UML	Unified Modeling Language
USML	United States Munitions List
V&V	Verification and Validation
VAC	Variance at Completion
VDHL	VHSIC Hardware Description Language
VHSIC	Very-High-Speed Integrated Circuit
WBS	Work Breakdown Structure
WP	Work Packages
WYE	Work Year Equivalent
XMI	XML Metadata Interchange
XML	Extensible Markup Language

Appendix B: Glossary

Term	Definition/Context
Acceptable Risk	The risk that is understood and agreed to by the program/project, governing authority, mission directorate, and other customer(s) such that no further specific mitigating action is required.
Acquisition	The process for obtaining the systems, research, services, construction, and supplies that NASA needs to fulfill its missions. Acquisition, which may include procurement (contracting for products and services), begins with an idea or proposal that aligns with the NASA Strategic Plan and fulfills an identified need and ends with the completion of the program or project or the final disposition of the product or service.
Activity	A set of tasks that describe the technical effort to accomplish a process and help generate expected outcomes.
Advancement Degree of Difficulty Assessment (AD2)	The process to develop an understanding of what is required to advance the level of system maturity.
Allocated Baseline (Phase C)	The allocated baseline is the approved performance-oriented configuration documentation for a CI to be developed that describes the functional and interface characteristics that are allocated from a higher level requirements document or a CI and the verification required to demonstrate achievement of those specified characteristics. The allocated baseline extends the top-level performance requirements of the functional baseline to sufficient detail for initiating manufacturing or coding of a CI. The allocated baseline is controlled by NASA. The allocated baseline(s) is typically established at the Preliminary Design Review.
Analysis	Use of mathematical modeling and analytical techniques to predict the compliance of a design to its requirements based on calculated data or data derived from lower system structure end product validations.
Analysis of Alternatives	A formal analysis method that compares alternative approaches by estimating their ability to satisfy mission requirements through an effectiveness analysis and by estimating their life-cycle costs through a cost analysis. The results of these two analyses are used together to produce a cost-effectiveness comparison that allows decision makers to assess the relative value or potential programmatic returns of the alternatives. An analysis of alternatives broadly examines multiple elements of program or project alternatives (including technical performance, risk, LCC, and programmatic aspects).
Analytic Hierarchy Process	A multi-attribute methodology that provides a proven, effective means to deal with complex decision- making and can assist with identifying and weighting selection criteria, analyzing the data collected for the criteria, and expediting the decision-making process.
Anomaly	The unexpected performance of intended function.
Approval	Authorization by a required management official to proceed with a proposed course of action. Approvals are documented.

Term	Definition/Context
Approval (for Implementation)	The acknowledgment by the decision authority that the program/project has met stakeholder expectations and formulation requirements, and is ready to proceed to implementation. By approving a program/project, the decision authority commits the budget resources necessary to continue into implementation. Approval (for Implementation) is documented.
Architecture (System)	Architecture is the high-level unifying structure that defines a system. It provides a set of rules, guidelines, and constraints that defines a cohesive and coherent structure consisting of constituent parts, relationships and connections that establish how those parts fit and work together. It addresses the concepts, properties and characteristics of the system and is represented by entities such as functions, functional flows, interfaces, relationships, resource flow items, physical elements, containers, modes, links, communication resources, etc. The entities are not independent but interrelated in the architecture through the relationships between them (NASA HQ). Fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution (ISO 42010).
As-Deployed Baseline	The as-deployed baseline occurs at the Operational Readiness Review. At this point, the design is considered to be functional and ready for flight. All changes will have been incorporated into the documentation.
Automated	Automation refers to the allocation of system functions to machines (hardware or software) versus humans.
Autonomous	Autonomy refers to the relative locations and scope of decision-making and control functions between two locations within a system or across the system boundary.
Baseline	An agreed-to set of requirements, designs, or documents that will have changes controlled through a formal approval and monitoring process.
Bidirectional Traceability	The ability to trace any given requirement/expectation to its parent requirement/expectation and to its allocated children requirements / expectations.
Brassboard	A medium fidelity functional unit that typically tries to make use of as much operational hardware/software as possible and begins to address scaling issues associated with the operational system. It does not have the engineering pedigree in all aspects, but is structured to be able to operate in simulated operational environments in order to assess performance of critical functions.
Breadboard	A low fidelity unit that demonstrates function only, without respect to form or fit in the case of hardware, or platform in the case of software. It often uses commercial and/or ad hoc components and is not intended to provide definitive information regarding operational performance.

Term	Definition/Context
Component Facilities	Complexes that are geographically separated from the NASA Center or institution to which they are assigned, but are still part of the Agency.
Concept of Operations (ConOps) (Concept Documentation)	Developed early in Pre-Phase A, the ConOps describes the overall high-level concept of how the system will be used to meet stakeholder expectations, usually in a time-sequenced manner. It describes the system from an operational perspective and helps facilitate an understanding of the system goals. It stimulates the development of the requirements and architecture related to the user elements of the system. It serves as the basis for subsequent definition documents and provides the foundation for the long-range operational planning activities.
Concurrence	A documented agreement by a management official that a proposed course of action is acceptable.
Concurrent Engineering	Design in parallel rather than serial engineering fashion. It is an approach to product development that brings manufacturing, testing, assurance, operations and other disciplines into the design cycle to ensure all aspects are incorporated into the design and thus reduce overall product development time.
Configuration Items	Any hardware, software, or combination of both that satisfies an end use function and is designated for separate configuration management. For example, configuration items can be referred to by an alphanumeric identifier which also serves as the unchanging base for the assignment of serial numbers to uniquely identify individual units of the CI.
Configuration Management Process	A management discipline that is applied over a product's life cycle to provide visibility into and to control changes to performance and functional and physical characteristics. It ensures that the configuration of a product is known and reflected in product information, that any product change is beneficial and is effected without adverse consequences, and that changes are managed.
Context Diagram	A diagram that shows external systems that impact the system being designed.
Continuous Risk Management	A systematic and iterative process that efficiently identifies, analyzes, plans, tracks, controls, communicates, and documents risks associated with implementation of designs, plans, and processes.
Contract	A mutually binding legal relationship obligating the seller to furnish the supplies or services (including construction) and the buyer to pay for them. It includes all types of commitments that obligate the Government to an expenditure of appropriated funds and that, except as otherwise authorized, are in writing. In addition to bilateral instruments, contracts include (but are not limited to) awards and notices of awards; job orders or task letters issued under basic ordering agreements; letter contracts; orders, such as purchase orders under which the contract becomes effective by written acceptance or performance; and bilateral contract modifications. Contracts do not include grants and cooperative agreements.

Term	Definition/Context
Contractor	An individual, partnership, company, corporation, association, or other service having a contract with the Agency for the design, development, manufacture, maintenance, modification, operation, or supply of items or services under the terms of a contract to a program or project. Research grantees, research contractors, and research subcontractors are excluded from this definition.
Control Account Manager	A manager responsible for a control account and for the planning, development, and execution of the budget content for those accounts.
Control Gate (or milestone)	A defined point in the program/project life cycle where the decision authority can evaluate progress and determine next actions. These may include a key decision point, life-cycle review, or other milestones identified by the program/project.
Cost-Benefit Analysis	A methodology to determine the advantage of one alternative over another in terms of equivalent cost or benefits. It relies on totaling positive factors and subtracting negative factors to determine a net result.
Cost-Effectiveness Analysis	A systematic quantitative method for comparing the costs of alternative means of achieving the same equivalent benefit for a specific objective.
Critical Design Review	A review that demonstrates that the maturity of the design is appropriate to support proceeding with full-scale fabrication, assembly, integration, and test, and that the technical effort is on track to complete the system development meeting performance requirements within the identified cost and schedule constraints.
Critical Event (or key event)	An event in the operations phase of the mission that is time-sensitive and is required to be accomplished successfully in order to achieve mission success. These events should be considered early in the life cycle as drivers for system design.
Critical Event Readiness Review	A review that evaluates the readiness of a project's flight system to execute the critical event during flight operation.
Customer	The organization or individual that has requested a product and will receive the product to be delivered. The customer may be an end user of the product, the acquiring agent for the end user, or the requestor of the work products from a technical effort. Each product within the system hierarchy has a customer.
Data Management	DM is used to plan for, acquire, access, manage, protect, and use data of a technical nature to support the total life cycle of a system.
Decision Analysis Process	A methodology for making decisions that offers techniques for modeling decision problems mathematically and finding optimal decisions numerically. The methodology entails identifying alternatives, one of which should be decided upon; possible events, one of which occurs thereafter; and outcomes, each of which results from a combination of decision and event.
Decision Authority	The individual authorized by the Agency to make important decisions for programs and projects under his or her authority.

Term	Definition/Context
Decision Matrix	A methodology for evaluating alternatives in which valuation criteria are typically displayed in rows on the left side of the matrix and alternatives are the column headings of the matrix. A “weight” is typically assigned to each criterion.
Decision Support Package	Documentation submitted in conjunction with formal reviews and change requests.
Decision Tree	A decision model that displays the expected consequences of all decision alternatives by making discreet all “chance” nodes, and, based on this, calculating and appropriately weighting the possible consequences of all alternatives.
Decommissioning Review	A review that confirms the decision to terminate or decommission a system and assess the readiness for the safe decommissioning and disposal of system assets. The DR is normally held near the end of routine mission operations upon accomplishment of planned mission objectives. It may be advanced if some unplanned event gives rise to a need to prematurely terminate the mission, or delayed if operational life is extended to permit additional investigations.
Deliverable Data Item	Consists of technical data, such as requirements specifications, design documents, management data plans, and metrics reports, that have been identified as items to be delivered with an end product.
Demonstration	Showing that the use of an end product achieves the individual specified requirement (verification) or stakeholder expectation (validation). It is generally a basic confirmation of performance capability, differentiated from testing by the lack of detailed data gathering. Demonstrations can involve the use of physical models or mockups; for example, a requirement that all controls shall be reachable by the pilot could be verified by having a pilot perform flight-related tasks in a cockpit mockup or simulator. A demonstration could also be the actual operation of the end product by highly qualified personnel, such as test pilots, who perform a one-time event that demonstrates a capability to operate at extreme limits of system performance.
Derived Requirements	Requirements arising from constraints, consideration of issues implied but not explicitly stated in the high-level direction provided by NASA Headquarters and Center institutional requirements, factors introduced by the selected architecture, and the design. These requirements are finalized through requirements analysis as part of the overall systems engineering process and become part of the program or project requirements baseline. Requirements arising from constraints, consideration of issues implied but not explicitly stated in the high-level direction provided by NASA Headquarters and Center institutional requirements, factors introduced by the selected architecture, and the design. These requirements are finalized through requirements analysis as part of the overall systems engineering process and become part of the program or project requirements baseline.
Descope	As a verb, take out of (or remove from) the scope of a project. As a noun, as in “performance descope,” it indicates the process or the result of the process of narrowing the scope; i.e., removing part of the original scope.

Term	Definition/Context
Design Solution Definition Process	The process used to translate the outputs of the logical decomposition process into a design solution definition. It includes transforming the defined logical decomposition models and their associated sets of derived technical requirements into alternative solutions and analyzing each alternative to be able to select a preferred alternative and fully define that alternative into a final design solution that will satisfy the technical requirements.
Designated Governing Authority	For the technical effort, this is the Center Director or the person that has been designated by the Center Director to ensure the appropriate level of technical management oversight. For large programs, this will typically be the Engineering Technical Authority. For smaller projects, this function can be delegated to line managers.
Detection	Determination that system state or behavior is different from expected performance.
Diagnosis	Determining the possible locations and/or causes of an anomaly or a failure.
Discrepancy	Any observed variance from, lack of agreement with, or contradiction to the required or expected outcome, configuration, or result.
Earned Value	The sum of the budgeted cost for tasks and products that have actually been produced (completed or in progress) at a given time in the schedule.
Earned Value Management	A tool for measuring and assessing project performance through the integration of technical scope with schedule and cost objectives during the execution of the project. EVM provides quantification of technical progress, enabling management to gain insight into project status and project completion costs and schedules. Two essential characteristics of successful EVM are EVM system data integrity and carefully targeted monthly EVM data analyses (i.e., risky WBS elements).
Emergent Behavior	An unanticipated behavior shown by a system due to interactions between a large numbers of simple components of that system.
End Product	The hardware/software or other product that performs the operational functions. This product is to be delivered to the next product layer or to the final customer.
Enabling Products	The life-cycle support products and services (e.g., production, test, deployment, training, maintenance, and disposal) that facilitate the progression and use of the operational end product through its life cycle. Since the end product and its enabling products are interdependent, they are viewed as a system. Project responsibility thus extends to acquiring services from the relevant enabling products in each life-cycle phase. When a suitable enabling product does not already exist, the project that is responsible for the end product may also be responsible for creating and using the enabling product.

Term	Definition/Context
Engineering Unit	A high fidelity unit that demonstrates critical aspects of the engineering processes involved in the development of the operational unit. Engineering test units are intended to closely resemble the final product (hardware/software) to the maximum extent possible and are built and tested so as to establish confidence that the design will function in the expected environments. In some cases, the engineering unit will become the final product, assuming that proper traceability has been exercised over the components and hardware handling.
Enhanced Functional Flow Block Diagram	A block diagram that represents control flows and data flows as well as system functions and flow.
Entrance Criteria	Guidance for minimum accomplishments each project needs to fulfill prior to a life-cycle review.
Environmental Impact	The direct, indirect, or cumulative beneficial or adverse effect of an action on the environment.
Environmental Management	The activity of ensuring that program and project actions and decisions that potentially impact or damage the environment are assessed and evaluated during the formulation and planning phase and reevaluated throughout implementation. This activity is performed according to all NASA policy and Federal, state, and local environmental laws and regulations.
Establish (with respect to processes)	The act of developing policy, work instructions, or procedures to implement process activities.
Evaluation	The continual self- and independent assessment of the performance of a program or project and incorporation of the evaluation findings to ensure adequacy of planning and execution according to plan.
Extensibility	The ability of a decision to be extended to other applications.
Failure	The inability of a system, subsystem, component, or part to perform its required function within specified limits (Source - NPR 8715.3 and Avizienis 2004).
Failure Tolerance	The ability to sustain a certain number of failures and still retain capability (Source – NPR 8705.2). A function should be preserved despite the presence of any of a specified number of coincident, independent failure causes of specified types.
Fault	A physical or logical cause, which explains a failure (Source – Avizienis 2004).
Fault Identification	Determining the possible locations of a failure or anomaly cause(s), to a defined level of granularity.
Fault Isolation	The act of containing the effects of a fault to limit the extent of failure.
Fault Management	A specialty engineering discipline that encompasses practices that enable an operational system to contain, prevent, detect, diagnose, identify, respond to, and recover from conditions that may interfere with nominal mission operations.

Term	Definition/Context
Fault Tolerance	See “failure tolerance.”
Feasible	Initial evaluations show that the concept credibly falls within the technical cost and schedule constraints for the project.
Flexibility	The ability of a decision to support more than one current application.
Flight Readiness Review	A review that examines tests, demonstrations, analyses, and audits that determine the system’s readiness for a safe and successful flight/launch and for subsequent flight operations. It also ensures that all flight and ground hardware, software, personnel, and procedures are operationally ready.
Float	The amount of time that a task in a project network schedule can be delayed without causing a delay to subsequent tasks or the project completion date.
Formulation Phase	The first part of the NASA management life cycle defined in NPR 7120.5 where system requirements are baselined, feasible concepts are determined, a system definition is baselined for the selected concept(s), and preparation is made for progressing to the Implementation Phase.
Functional Analysis	The process of identifying, describing, and relating the functions a system should perform to fulfill its goals and objectives.
Functional Baseline (Phase B)	The functional baseline is the approved configuration documentation that describes a system’s or top-level CIs’ performance requirements (functional, interoperability, and interface characteristics) and the verification required to demonstrate the achievement of those specified characteristics.
Functional Configuration Audit (FCA)	Examines the functional characteristics of the configured product and verifies that the product has met, via test results, the requirements specified in its functional baseline documentation approved at the PDR and CDR plus any approved changes thereafter. FCAs will be conducted on both hardware- and software-configured products and will precede the PCA of the configured product.
Functional Decomposition	A subfunction under logical decomposition and design solution definition, it is the examination of a function to identify subfunctions necessary for the accomplishment of that function and functional relationships and interfaces.
Functional Flow Block Diagram	A block diagram that defines system functions and the time sequence of functional events.
Gantt Chart	A bar chart depicting start and finish dates of activities and products in the WBS.
Goal	Goals elaborate on the need and constitute a specific set of expectations for the system. They further define what we hope to accomplish by addressing the critical issues identified during the problem assessment. Goals need not be in a quantitative or measurable form, but they must allow us to assess whether the system has achieved them.

Term	Definition/Context
Government Mandatory Inspection Points	Inspection points required by Federal regulations to ensure 100 percent compliance with safety/mission-critical attributes when noncompliance can result in loss of life or loss of mission.
Health Assessment	The activity under Fault Management that carries out detection, diagnosis, and identification of faults and prediction of fault propagation states into the future.
Health Monitoring	The activity under Fault Management that implements system state data collection, storage, and reporting though sensing and communication.
Heritage (or legacy)	Refers to the original manufacturer's level of quality and reliability that is built into the parts, which have been proven by (1) time in service, (2) number of units in service, (3) mean time between failure performance, and (4) number of use cycles.
Human-Centered Design (HCD)	An approach to the development of interactive systems that focuses on making systems usable by ensuring that the needs, abilities, and limitations of the human user are met throughout the system's life cycle.
Human Factors Engineering	The discipline that studies human-system interfaces and provides requirements, standards, and guidelines to ensure the human component of an integrated system is able to function as intended.
Human Systems Integration (HSI)	An interdisciplinary and comprehensive management and technical process that focuses on the integration of human considerations into the system acquisition and development processes to enhance human system design, reduce life-cycle ownership cost, and optimize total system performance.
Implementation Phase	The part of the NASA management life cycle defined in NPR 7120.5 where the detailed design of system products is completed and the products to be deployed are fabricated, assembled, integrated, and tested and the products are deployed to their customers or users for their assigned use or mission.
Incommensurable Costs	Costs that cannot be easily measured, such as controlling pollution on launch or mitigating debris.
Influence Diagram	A compact graphical and mathematical representation of a decision state. Its elements are decision nodes, chance nodes, value nodes, and arrows to indicate the relationships among these elements.
Inspection	The visual examination of a realized end product. Inspection is generally used to verify physical design features or specific manufacturer identification. For example, if there is a requirement that the safety arming pin has a red flag with the words "Remove Before Flight" stenciled on the flag in black letters, a visual inspection of the arming pin flag can be used to determine if this requirement was met.

Term	Definition/Context
Integrated Logistics Support	The management, engineering activities, analysis, and information management associated with design requirements definition, material procurement and distribution, maintenance, supply replacement, transportation, and disposal that are identified by space flight and ground systems supportability objectives.
Interface Management Process	The process to assist in controlling product development when efforts are divided among parties (e.g., Government, contractors, geographically diverse technical teams) and/or to define and maintain compliance among the products that should interoperate.
Iterative	Application of a process to the same product or set of products to correct a discovered discrepancy or other variation from requirements. (See “recursive” and “repeatable.”)
Key Decision Point	The event at which the decision authority determines the readiness of a program/project to progress to the next phase of the life cycle (or to the next KDP).
Key Event (or Critical Event)	See “critical event.”
Key Performance Parameter	Those capabilities or characteristics (typically engineering-based or related to health and safety or operational performance) considered most essential for successful mission accomplishment. They characterize the major drivers of operational performance, supportability, and interoperability.
Knowledge Management	A collection of policies, processes, and practices relating to the use of intellectual- and knowledge-based assets in an organization.
Least-Cost Analysis	A methodology that identifies the least-cost project option for meeting the technical requirements.
Liens	Requirements or tasks not satisfied that have to be resolved within a certain assigned time to allow passage through a control gate to proceed.
Life-Cycle Cost	The total of the direct, indirect, recurring, nonrecurring, and other related expenses both incurred and estimated to be incurred in the design, development, verification, production, deployment, prime mission operation, maintenance, support, and disposal of a project, including closeout, but not extended operations. The LCC of a project or system can also be defined as the total cost of ownership over the project or system’s planned life cycle from Formulation (excluding Pre–Phase A) through Implementation (excluding extended operations). The LCC includes the cost of the launch vehicle.
Logical Decomposition Models	Mathematical or visual representations of the relationships between requirements as identified in the Logical Decomposition Process.

Term	Definition/Context
Logical Decomposition Process	A process used to improve understanding of the defined technical requirements and the relationships among the requirements (e.g., functional, behavioral, performance, and temporal) and to transform the defined set of technical requirements into a set of logical decomposition models and their associated set of derived technical requirements for lower levels of the system and for input to the Design Solution Definition Process.
Logistics (or Integrated Logistics Support)	See “integrated logistics support.”
Loosely Coupled Program	Programs that address specific objectives through multiple space flight projects of varied scope. While each individual project has an assigned set of mission objectives, architectural and technological synergies and strategies that benefit the program as a whole are explored during the formulation process. For instance, Mars orbiters designed for more than one Mars year in orbit are required to carry a communication system to support present and future landers.
Maintain (with respect to establishment of processes)	The act of planning the process, providing resources, assigning responsibilities, training people, managing configurations, identifying and involving stakeholders, and monitoring process effectiveness.
Maintainability	The measure of the ability of an item to be retained in or restored to specified conditions when maintenance is performed by personnel having specified skill levels, using prescribed procedures and resources, at each prescribed level of maintenance.
Margin	The allowances carried in budget, projected schedules, and technical performance parameters (e.g., weight, power, or memory) to account for uncertainties and risks. Margins are allocated in the formulation process based on assessments of risks and are typically consumed as the program/ project proceeds through the life cycle.
Master Equipment List	The Master Equipment List (MEL) is a listing of all the parts of a system and includes pertinent information such as serial numbers, model numbers, manufacturer, equipment type, system/element it is located within, etc.
Measure of Effectiveness	A measure by which a stakeholder’s expectations are judged in assessing satisfaction with products or systems produced and delivered in accordance with the associated technical effort. The MOE is deemed to be critical to not only the acceptability of the product by the stakeholder but also critical to operational/mission usage. A MOE is typically qualitative in nature or not able to be used directly as a design-to requirement.
Measure of Performance	A quantitative measure that, when met by the design solution, helps ensure that a MOE for a product or system will be satisfied. These MOPs are given special attention during design to ensure that the MOEs to which they are associated are met. There are generally two or more measures of performance for each MOE.
Metric	The result of a measurement taken over a period of time that communicates vital information about the status or performance of a system, process, or activity. A metric should drive appropriate action.

Term	Definition/Context
Mission	A major activity required to accomplish an Agency goal or to effectively pursue a scientific, technological, or engineering opportunity directly related to an Agency goal. Mission needs are independent of any particular system or technological solution.
Mission Concept Review	A review that affirms the mission/project need and examines the proposed mission's objectives and the ability of the concept to fulfill those objectives.
Mission Definition Review	A life-cycle review that evaluates whether the proposed mission/system architecture is responsive to the program mission/system functional and performance requirements and requirements have been allocated to all functional elements of the mission/system.
Mitigation	An action taken to mitigate the effects of a fault towards achieving existing or redefined system goals.
Model	A model is a physical, mathematical, or logical representation of reality.
Need	A single statement that drives everything else. It should relate to the problem that the system is supposed to solve, but not be the solution.
Nonconforming product	Software, hardware, or combination, either produced, acquired, or in some combination that is identified as not meeting documented requirements.
Objective	Specific target levels of outputs the system must achieve. Each objective should relate to a particular goal. Generally, objectives should meet four criteria: (1) Specific - Objectives should aim at results and reflect what the system needs to do, but they don't outline how to implement the solution. They need to be specific enough to provide clear direction, so developers, customers, and testers can understand them. (2) Measurable - Objectives need to be quantifiable and verifiable. The project needs to monitor the system's success in achieving each objective. (3) Aggressive, but attainable- Objectives need to be challenging but reachable, and targets need to be realistic. At first, objectives "To Be Determined" (TBD) may be included until trade studies occur, operations concepts solidify, or technology matures. But objectives need to be feasible before starting to write requirements and design systems. (4) Results-oriented - Objectives need to focus on desired outputs and outcomes, not on the methods used to achieve the target (what, not how).
Objective Function (sometimes Cost Function)	A mathematical expression of the values of combinations of possible outcomes as a single measure of cost-effectiveness.
Operational Environment	The environment in which the final product will be operated. In the case of space flight hardware/software, it is space. In the case of ground-based or airborne systems that are not directed toward space flight, it is the environments defined by the scope of operations. For software, the environment is defined by the operational platform.

Term	Definition/Context
Operational Readiness Review	A review that examines the actual system characteristics and the procedures used in the system or product's operation and ensures that all system and support (flight and ground) hardware, software, personnel, procedures, and user documentation accurately reflects the deployed state of the system and are operationally ready.
Operations Concept	A description of how the flight system and the ground system are used together to ensure that the concept of operation is reasonable. This might include how mission data of interest, such as engineering or scientific data, are captured, returned to Earth, processed, made available to users, and archived for future reference. (Source - NPR 7120.5)
Optimal Solution	A feasible solution that best meets criteria when balanced at a system level.
Other Interested Parties (Stakeholders)	A subset of "stakeholders," other interested parties are groups or individuals who are not customers of a planned technical effort but may be affected by the resulting product, the manner in which the product is realized or used, or have a responsibility for providing life-cycle support services.
Peer Review	Independent evaluation by internal or external subject matter experts who do not have a vested interest in the work product under review. Peer reviews can be planned, focused reviews conducted on selected work products by the producer's peers to identify defects and issues prior to that work product moving into a milestone review or approval cycle.
Performance Standards	Defines what constitutes acceptable performance by the provider. Common metrics for use in performance standards include cost and schedule.
Physical Configuration Audits (or configuration inspection)	The PCA examines the physical configuration of the configured product and verifies that the product corresponds to the build-to (or code-to) product baseline documentation previously approved at the CDR plus the approved changes thereafter. PCAs are conducted on both hardware-and software-configured products.
Post-Flight Assessment Review	Evaluates how well mission objectives were met during a mission and identifies all flight and ground system anomalies that occurred during the flight and determines the actions necessary to mitigate or resolve the anomalies for future flights of the same spacecraft design.
Post-Launch Assessment Review	A review that evaluates the readiness of the spacecraft systems to proceed with full, routine operations after post-launch deployment. The review also evaluates the status of the project plans and the capability to conduct the mission with emphasis on near-term operations and mission-critical events.
Precedence Diagram	Workflow diagram that places activities in boxes connected by dependency arrows; typical of a Gantt chart.

Term	Definition/Context
Preliminary Design Review	A review that demonstrates that the preliminary design meets all system requirements with acceptable risk and within the cost and schedule constraints and establishes the basis for proceeding with detailed design. It will show that the correct design option has been selected, interfaces have been identified, and verification methods have been described.
Process	A set of activities used to convert inputs into desired outputs to generate expected outcomes and satisfy a purpose.
Producibility	A system characteristic associated with the ease and economy with which a completed design can be transformed (i.e., fabricated, manufactured, or coded) into a hardware and/or software realization.
Product	A part of a system consisting of end products that perform operational functions and enabling products that perform life-cycle services related to the end product or a result of the technical efforts in the form of a work product (e.g., plan, baseline, or test result).
Product Baseline (Phase D/E)	The product baseline is the approved technical documentation that describes the configuration of a CI during the production, fielding / deployment, and operational support phases of its life cycle. The product baseline describes detailed physical or form, fit, and function characteristics of a CI; the selected functional characteristics designated for production acceptance testing; and the production acceptance test requirements.
Product Breakdown Structure	A hierarchical breakdown of the hardware and software products of a program/project.
Product Implementation Process	A process used to generate a specified product of a product layer through buying, making, or reusing in a form consistent with the product life-cycle phase exit (success) criteria and that satisfies the design solution definition-specified requirements (e.g., drawings, specifications).
Product Integration Process	A process used to transform the design solution definition into the desired end product of the product layer through assembly and integration of lower-level validated end products in a form that is consistent with the product life-cycle phase exit (success) criteria and that satisfies the design solution definition requirements (e.g., drawings, specifications).
Product Realization	The act of making, buying, or reusing a product, or the assembly and integration of lower-level realized products into a new product, as well as the verification and validation that the product satisfies its appropriate set of requirements and the transition of the product to its customer.
Product Transition Process	A process used to transition a verified and validated end product that has been generated by product implementation or product integration to the customer at the next level in the system structure for integration into an end product or, for the top-level end product, transitioned to the intended end user.

Term	Definition/Context
Product Validation Process	A process used to confirm that a verified end product generated by product implementation or product integration fulfills (satisfies) its intended use when placed in its intended environment and to assure that any anomalies discovered during validation are appropriately resolved prior to delivery of the product (if validation is done by the supplier of the product) or prior to integration with other products into a higher-level assembled product (if validation is done by the receiver of the product). The validation is done against the set of baselined stakeholder expectations.
Product Verification Process	A process used to demonstrate that an end product generated from product implementation or product integration conforms to its design solution definition requirements as a function of the product life-cycle phase and the location of the product layer end product in the system structure.
Production Readiness Review	A review for projects developing or acquiring multiple or similar systems greater than three or as determined by the project. The PRR determines the readiness of the system developers to efficiently produce the required number of systems. It ensures that the production plans, fabrication, assembly, integration-enabling products, operational support, and personnel are in place and ready to begin production.
Prognosis	The prediction of a system's future health states, degradation, and Remaining Useful Life (RUL).
Program	A strategic investment by a mission directorate or mission support office that has a defined architecture and/or technical approach, requirements, funding level, and a management structure that initiates and directs one or more projects. A program defines a strategic direction that the Agency has identified as critical.
Program/System Definition Review	A review that examines the proposed program architecture and the flowdown to the functional elements of the system. The proposed program's objectives and the concept for meeting those objectives are evaluated. Key technologies and other risks are identified and assessed. The baseline program plan, budgets, and schedules are presented.
Program Requirements	The set of requirements imposed on the program office, which are typically found in the program plan plus derived requirements that the program imposes on itself.
Program System Requirements Review	A review that evaluates the credibility and responsiveness of a proposed program requirements/architecture to the mission directorate requirements, the allocation of program requirements to the projects, and the maturity of the program's mission/system definition.
Programmatic Requirements	Requirements set by the mission directorate, program, project, and PI, if applicable. These include strategic scientific and exploration requirements, system performance requirements, and schedule, cost, and similar nontechnical constraints.

Term	Definition/Context
Project	A specific investment having defined goals, objectives, requirements, life-cycle cost, a beginning, and an end. A project yields new or revised products or services that directly address NASA's strategic needs. The products may be produced or the services performed wholly in-house; by partnerships with Government, industry, or academia; or through contracts with private industry.
Project Plan	The document that establishes the project's baseline for implementation, signed by the responsible program manager, Center Director, project manager, and the MDAA, if required.
Project Requirements	The set of requirements imposed on the project and developer, which are typically found in the project plan plus derived requirements that the project imposes on itself. It includes identification of activities and deliverables (end products and work products) and outputs of the development and operations.
Phase Product	An end product that is to be provided as a result of the activities of a given life-cycle phase. The form depends on the phase – a product of early phases might be a simulation or model; a product of later phases may be the (final) end product itself.
Product Form	A representation of a product that depends on the development phase, current use, and maturity. Examples include mockup, model, engineering unit, prototype unit, and flight unit.
Product Realization	The desired output from the application of the four product realization processes. The form of this product is dependent on the phase of the product life cycle and the phase exit (success) criteria.
Prototype	The prototype unit demonstrates form, fit, and function at a scale deemed to be representative of the final product operating in its operational environment. A subscale test article provides fidelity sufficient to permit validation of analytical models capable of predicting the behavior of full-scale systems in an operational environment. The prototype is used to "wring out" the design solution so that experience gained from the prototype can be fed back into design changes that will improve the manufacture, integration, and maintainability of a single flight item or the production run of several flight items.
Quality Assurance	An independent assessment performed throughout a product's life cycle in order to acquire confidence that the system actually produced and delivered is in accordance with its functional, performance, and design requirements.
Realized Product	The end product that has been implemented / integrated, verified, validated, and transitioned to the next product layer.
Recovery	An action taken to restore the functions necessary to achieve existing or redefined system goals after a fault/failure occurs.

Term	Definition/Context
Recursive	Value is added to the system by the repeated application of processes to design next lower-layer system products or to realize next upper-layer end products within the system structure. This also applies to repeating the application of the same processes to the system structure in the next life-cycle phase to mature the system definition and satisfy phase exit (success) criteria.
Relevant Stakeholder	A subset of the term “stakeholder” that applies to people or roles that are designated in a plan for stakeholder involvement. Since “stakeholder” may describe a very large number of people, a lot of time and effort would be consumed by attempting to deal with all of them. For this reason, “relevant stakeholder” is used in most practice statements to describe the people identified to contribute to a specific task.
Relevant Environment	Not all systems, subsystems, and/or components need to be operated in the operational environment in order to satisfactorily address performance margin requirements or stakeholder expectations. Consequently, the relevant environment is the specific subset of the operational environment that is required to demonstrate critical “at risk” aspects of the final product performance in an operational environment.
Reliability	The measure of the degree to which a system ensures mission success by functioning properly over its intended life. It has a low and acceptable probability of failure, achieved through simplicity, proper design, and proper application of reliable parts and materials. In addition to long life, a reliable system is robust and fault tolerant.
Repeatable	A characteristic of a process that can be applied to products at any level of the system structure or within any life-cycle phase.
Requirement	The agreed-upon need, desire, want, capability, capacity, or demand for personnel, equipment, facilities, or other resources or services by specified quantities for specific periods of time or at a specified time expressed as a “shall” statement. Acceptable form for a requirement statement is individually clear, correct, feasible to obtain, unambiguous in meaning, and can be validated at the level of the system structure at which it is stated. In pairs of requirement statements or as a set, collectively, they are not redundant, are adequately related with respect to terms used, and are not in conflict with one another.
Requirements Allocation Sheet	Documents the connection between allocated functions, allocated performance, and the physical system.
Requirements Management Process	A process used to manage the product requirements identified, baselined, and used in the definition of the products of each product layer during system design. It provides bidirectional traceability back to the top product layer requirements and manages the changes to established requirement baselines over the life cycle of the system products.

Term	Definition/Context
Risk	In the context of mission execution, risk is the potential for performance shortfalls that may be realized in the future with respect to achieving explicitly established and stated performance requirements. The performance shortfalls may be related to any one or more of the following mission execution domains: (1) safety, (2) technical, (3) cost, and (4) schedule. (Source - NPR 8000.4, Agency Risk Management Procedural Requirements)
Risk Assessment	An evaluation of a risk item that determines (1) what can go wrong, (2) how likely it is to occur,(3) what the consequences are, and (4) what the uncertainties associated with the likelihood and consequences are, and 5) what the mitigation plans are.
Risk-Informed Decision Analysis Process	A five-step process focusing first on objectives and next on developing decision alternatives with those objectives clearly in mind and/or using decision alternatives that have been developed under other systems engineering processes. The later steps of the process interrelate heavily with the Technical Risk Management Process.
Risk Management	Risk management includes Risk-Informed Decision-Making (RIDM) and Continuous Risk Management (CRM) in an integrated framework. RIDM informs systems engineering decisions through better use of risk and uncertainty information in selecting alternatives and establishing baseline requirements. CRM manages risks over the course of the development and the Implementation Phase of the life cycle to ensure that safety, technical, cost, and schedule requirements are met. This is done to foster proactive risk management, to better inform decision-making through better use of risk information, and then to more effectively manage Implementation risks by focusing the CRM process on the baseline performance requirements emerging from the RIDM process. (Source- NPR 8000.4, Agency Risk Management Procedural Requirements) These processes are applied at a level of rigor commensurate with the complexity, cost, and criticality of the program.
Safety	Freedom from those conditions that can cause death, injury, occupational illness, damage to or loss of equipment or property, or damage to the environment.
Search Space (or Alternative Space)	The envelope of concept possibilities defined by design constraints and parameters within which alternative concepts can be developed and traded off.
Single-Project Programs	Programs that tend to have long development and/or operational lifetimes, represent a large investment of Agency resources, and have contributions from multiple organizations/agencies. These programs frequently combine program and project management approaches, which they document through tailoring.

Term	Definition/Context
Software	Computer programs, procedures, rules, and associated documentation and data pertaining to the development and operation of a computer system. Software also includes Commercial Off-The-Shelf (COTS), Government Off-The-Shelf (GOTS), Modified Off-The-Shelf (MOTS), embedded software, reuse, heritage, legacy, autogenerated code, firmware, and open source software components. Note 1: For purposes of the NASA Software Release program only, the term "software," as redefined in NPR 2210.1, Release of NASA Software, does not include computer databases or software documentation. Note 2: Definitions for the terms COTS, GOTS, heritage software, MOTS, legacy software, software reuse, and classes of software are provided in NPR 7150.2, NASA Software Engineering Requirements. (Source - NPD 7120.4, NASA Engineering and Program/Project Management Policy)
Solicitation	The vehicle by which information is solicited from contractors for the purpose of awarding a contract for products or services. Any request to submit offers or quotations to the Government. Solicitations under sealed bid procedures are called “invitations for bids.” Solicitations under negotiated procedures are called “requests for proposals.” Solicitations under simplified acquisition procedures may require submission of either a quotation or an offer.
Specification	A document that prescribes completely, precisely, and verifiably the requirements, design, behavior, or characteristics of a system or system component. In NPR 7123.1, “specification” is treated as a “requirement.”
Stakeholder	A group or individual who is affected by or has an interest or stake in a program or project. There are two main classes of stakeholders. See "customers" and "other interested parties."
Stakeholder Expectations	A statement of needs, desires, capabilities, and wants that are not expressed as a requirement (not expressed as a “shall” statement) is referred to as an “expectation.” Once the set of expectations from applicable stakeholders is collected, analyzed, and converted into a “shall” statement, the expectation becomes a requirement. Expectations can be stated in either qualitative (nonmeasurable) or quantitative (measurable) terms. Requirements are always stated in quantitative terms. Expectations can be stated in terms of functions, behaviors, or constraints with respect to the product being engineered or the process used to engineer the product.
Stakeholder Expectations Definition Process	A process used to elicit and define use cases, scenarios, concept of operations, and stakeholder expectations for the applicable product life-cycle phases and product layer. The baselined stakeholder expectations are used for validation of the product layer end product.
Standing Review Board	The board responsible for conducting independent reviews (life-cycle and special) of a program or project and providing objective, expert judgments to the convening authorities. The reviews are conducted in accordance with approved Terms of Reference (ToR) and life-cycle requirements per NPR 7123.1.

Term	Definition/Context
State Diagram	A diagram that shows the flow in the system in response to varying inputs in order to characterize the behavior of the system.
Success Criteria	Specific accomplishments that need to be satisfactorily demonstrated to meet the objectives of a technical review so that a technical effort can progress further in the life cycle. Success criteria are documented in the corresponding technical review plan. Formerly referred to as “exit” criteria, a term still used in some NPDs/NPRs.
Surveillance	The monitoring of a contractor’s activities (e.g., status meetings, reviews, audits, site visits) for progress and production and to demonstrate fiscal responsibility, ensure crew safety and mission success, and determine award fees for extraordinary (or penalty fees for substandard) contract execution.
System	(1) The combination of elements that function together to produce the capability to meet a need. The elements include all hardware, software, equipment, facilities, personnel, processes, and procedures needed for this purpose. (2) The end product (which performs operational functions) and enabling products (which provide life-cycle support services to the operational end products) that make up a system.
System Acceptance Review	The SAR verifies the completeness of the specific end products in relation to their expected maturity level, assesses compliance to stakeholder expectations, and ensures that the system has sufficient technical maturity to authorize its shipment to the designated operational facility or launch site.
System Definition Review	The Mission / System Definition Review (MDR/SDR) evaluates whether the proposed mission/system architecture is responsive to the program mission/system functional and performance requirements and requirements have been allocated to all functional elements of the mission/system. This review is used for projects and for single-project programs.
System Integration Review	A SIR ensures that segments, components, and subsystems are on schedule to be integrated into the system and that integration facilities, support personnel, and integration plans and procedures are on schedule to support integration.
System Requirements Review	For a program, the SRR is used to ensure that its functional and performance requirements are properly formulated and correlated with the Agency and mission directorate strategic objectives. For a system/project, the SRR evaluates whether the functional and performance requirements defined for the system are responsive to the program's requirements and ensures that the preliminary project plan and requirements will satisfy the mission.
System Safety Engineering	The application of engineering and management principles, criteria, and techniques to achieve acceptable mishap risk within the constraints of operational effectiveness and suitability, time, and cost throughout all phases of the system life cycle.

Term	Definition/Context
System Structure	A system structure is made up of a layered structure of product-based WBS models. (See “Work Breakdown Structure” and Product Breakdown Structure.”)
Systems Approach	The application of a systematic, disciplined engineering approach that is quantifiable, recursive, iterative, and repeatable for the development, operation, and maintenance of systems integrated into a whole throughout the life cycle of a project or program.
Systems Engineering Engine	The SE model shown in Figure 2.1-1 that provides the 17 technical processes and their relationships with each other. The model is called an “SE engine” in that the appropriate set of processes is applied to the products being engineered to drive the technical effort.
Systems Engineering Management Plan	The SEMP identifies the roles and responsibility interfaces of the technical effort and specifies how those interfaces will be managed. The SEMP is the vehicle that documents and communicates the technical approach, including the application of the common technical processes; resources to be used; and the key technical tasks, activities, and events along with their metrics and success criteria.
Tailoring	A process used to adjust or seek relief from a prescribed requirement to accommodate the needs of a specific task or activity (e.g., program or project). The tailoring process results in the generation of deviations and waivers depending on the timing of the request. OR The process used to seek relief from NPR 7123.1 requirements consistent with program or project objectives, allowable risk, and constraints.
Technical Assessment Process	A process used to help monitor progress of the technical effort and provide status information for support of the system design, product realization, and technical management processes. A key aspect of the process is conducting life-cycle and technical reviews throughout the system life cycle.
Technical Cost Estimate	The cost estimate of the technical work on a project created by the technical team based on its understanding of the system requirements and operational concepts and its vision of the system architecture.
Technical Data Management Process	A process used to plan for, acquire, access, manage, protect, and use data of a technical nature to support the total life cycle of a system. This process is used to capture trade studies, cost estimates, technical analyses, reports, and other important information.
Technical Data Package	An output of the Design Solution Definition Process, it evolves from phase to phase, starting with conceptual sketches or models and ending with complete drawings, parts list, and other details needed for product implementation or product integration.

Term	Definition/Context
Technical Measures	An established set of measures based on the expectations and requirements that will be tracked and assessed to determine overall system or product effectiveness and customer satisfaction. Common terms for these measures are Measures Of Effectiveness (MOEs), Measures Of Performance (MOPs), and Technical Performance Measures (TPMs).
Technical Performance Measures	A set of performance measures that are monitored by comparing the current actual achievement of the parameters with that anticipated at the current time and on future dates. TPMs are used to confirm progress and identify deficiencies that might jeopardize meeting a system requirement. Assessed parameter values that fall outside an expected range around the anticipated values indicate a need for evaluation and corrective action. Technical performance measures are typically selected from the defined set of Measures Of Performance (MOPs).
Technical Planning Process	A process used to plan for the application and management of each common technical process. It is also used to identify, define, and plan the technical effort applicable to the product life-cycle phase for product layer location within the system structure and to meet project objectives and product life-cycle phase exit (success) criteria. A key document generated by this process is the SEMP.
Technical Requirements	A set of requirements imposed on the end products of the system, including the system itself. Also referred to as “product requirements.”
Technical Requirements Definition Process	A process used to transform the stakeholder expectations into a complete set of validated technical requirements expressed as “shall” statements that can be used for defining a design solution for the Product Breakdown Structure (PBS) model and related enabling products.
Technical Risk	Risk associated with the achievement of a technical goal, criterion, or objective. It applies to undesired consequences related to technical performance, human safety, mission assets, or environment.
Technical Risk Management Process	A process used to make risk-informed decisions and examine, on a continuing basis, the potential for deviations from the project plan and the consequences that could result should they occur.
Technical Team	A group of multidisciplinary individuals with appropriate domain knowledge, experience, competencies, and skills who are assigned to a specific technical task.
Technology Readiness Assessment Report	A document required for transition from Phase B to Phase C/D demonstrating that all systems, subsystems, and components have achieved a level of technological maturity with demonstrated evidence of qualification in a relevant environment.

Term	Definition/Context
Technology Assessment	A systematic process that ascertains the need to develop or infuse technological advances into a system. The technology assessment process makes use of basic systems engineering principles and processes within the framework of the Product Breakdown Structure (PBS). It is a two-step process comprised of (1) the determination of the current technological maturity in terms of Technology Readiness Levels (TRLs) and (2) the determination of the difficulty associated with moving a technology from one TRL to the next through the use of the Advancement Degree of Difficulty Assessment (AD^2).
Technology Development Plan	A document required for transition from Phase A to Phase B identifying technologies to be developed, heritage systems to be modified, alternative paths to be pursued, fallback positions and corresponding performance descopes, milestones, metrics, and key decision points. It is incorporated in the preliminary project plan.
Technology Maturity Assessment	A process to determine a system's technological maturity based on Technology Readiness Levels (TRLs).
Technology Readiness Level	Provides a scale against which to measure the maturity of a technology. TRLs range from 1, basic technology research, to 9, systems test, launch, and operations. Typically, a TRL of 6 (i.e., technology demonstrated in a relevant environment) is required for a technology to be integrated into an SE process.
Test	The use of a realized end product to obtain detailed data to verify or validate performance or to provide sufficient information to verify or validate performance through further analysis.
Test Readiness Review	A review that ensures that the test article (hardware/software), test facility, support personnel, and test procedures are ready for testing and data acquisition, reduction, and control.
Threshold Requirements	A minimum acceptable set of technical and project requirements; the set could represent the descope position of the project.
Tightly Coupled Programs	Programs with multiple projects that execute portions of a mission(s). No single project is capable of implementing a complete mission. Typically, multiple NASA Centers contribute to the program. Individual projects may be managed at different Centers. The program may also include contributions from other agencies or international partners.
Traceability	A discernible association among two or more logical entities such as requirements, system elements, verifications, or tasks.
Trade Study	A means of evaluating system designs by devising alternative means to meet functional requirements, evaluating these alternatives in terms of the measures of effectiveness and system cost, ranking the alternatives according to appropriate selection criteria, dropping less promising alternatives, and proceeding to the next level of resolution, if needed.

Term	Definition/Context
Trade Study Report	A report written to document a trade study. It should include: the system under analysis; system goals, objectives (or requirements, as appropriate to the level of resolution), and constraints; measures and measurement methods (models) used; all data sources used; the alternatives chosen for analysis; computational results, including uncertainty ranges and sensitivity analyses performed; the selection rule used; and the recommended alternative.
Trade Tree	A representation of trade study alternatives in which each layer represents some system aspect that needs to be treated in a trade study to determine the best alternative.
Transition	The act of delivery or moving of a product from one location to another. This act can include packaging, handling, storing, moving, transporting, installing, and sustainment activities.
Uncoupled Programs	Programs implemented under a broad theme and/or a common program implementation concept, such as providing frequent flight opportunities for cost-capped projects selected through AO or NASA Research Announcements. Each such project is independent of the other projects within the program.
Utility	A measure of the relative value gained from an alternative. The theoretical unit of measurement for utility is the “util.”
Validated Requirements	A set of requirements that are well formed (clear and unambiguous), complete (agree with customer and stakeholder needs and expectations), consistent (conflict free), and individually verifiable and traceable to a higher level requirement or goal.
Validation (of a product)	The process of showing proof that the product accomplishes the intended purpose based on stakeholder expectations and the Concept of Operations. May be determined by a combination of test, analysis, demonstration, and inspection. (Answers the question, “Am I building the right product?”)
Variance	In program control terminology, a difference between actual performance and planned costs or schedule status.
Verification (of a product)	Proof of compliance with specifications. Verification may be determined by test, analysis, demonstration, or inspection or a combination thereof. (Answers the question, “Did I build the product right?”)
Waiver	A documented authorization releasing a program or project from meeting a requirement after the requirement is put under configuration control at the level the requirement will be implemented.
WBS Model	A Work Breakdown Structure (WBS) model describes a system that consists of end products and their subsystems (which perform the operational functions of the system), the supporting or enabling products, and any other work products (plans, baselines) required for the development of the system.

Term	Definition/Context
Work Breakdown Structure (WBS)	A product-oriented hierarchical division of the hardware, software, services, and data required to produce the program/project's end product(s) structured according to the way the work will be performed, reflecting the way in which program/project costs, schedule, technical, and risk data are to be accumulated, summarized, and reported.
Workflow Diagram	A scheduling chart that shows activities, dependencies among activities, and milestones.

Appendix C: How to Write a Good Requirement - Checklist

C.1 Use of Correct Terms

- ☐ Shall = requirement
- ☐ Will = facts or declaration of purpose
- ☐ Should = goal

C.2 Editorial Checklist

Personnel Requirement

- ☐ The requirement is in the form "responsible party shall perform such and such." In other words, use the active, rather than the passive voice. A requirement should state who shall (do, perform, provide, weigh, or other verb) followed by a description of what should be performed.

Product Requirement

- ☐ The requirement is in the form "product ABC shall XYZ." A requirement should state "The product shall" (do, perform, provide, weigh, or other verb) followed by a description of what should be done.
- ☐ The requirement uses consistent terminology to refer to the product and its lower-level entities.
- ☐ Complete with tolerances for qualitative/performance values (e.g., less than, greater than or equal to, plus or minus, 3 sigma root sum squares).
- ☐ Is the requirement free of implementation? (Requirements should state WHAT is needed, NOT HOW to provide it; i.e., state the problem not the solution. Ask, "Why do you need the requirement?" The answer may point to the real requirement.)
- ☐ Free of descriptions of operations? (Is this a need the product should satisfy or an activity involving the product? Sentences like "The operator shall…" are almost always operational statements not requirements.)

Example Product Requirements

- The system shall operate at a power level of…
- The software shall acquire data from the…
- The structure shall withstand loads of…
- The hardware shall have a mass of…

C.3 General Goodness Checklist

- ☐ The requirement is grammatically correct.
- ☐ The requirement is free of typos, misspellings, and punctuation errors.
- ☐ The requirement complies with the project's template and style rules.
- ☐ The requirement is stated positively (as opposed to negatively, i.e., "shall not").

- ☐ The use of “To Be Determined” (TBD) values should be minimized. It is better to use a best estimate for a value and mark it “To Be Resolved” (TBR) with the rationale along with what should be done to eliminate the TBR, who is responsible for its elimination, and by when it should be eliminated.
- ☐ The requirement is accompanied by an intelligible rationale, including any assumptions. Can you validate (concur with) the assumptions? Assumptions should be confirmed before baselining.
- ☐ The requirement is located in the proper section of the document (e.g., not in an appendix).

C.4 Requirements Validation Checklist

Clarity

- ☐ Are the requirements clear and unambiguous? (Are all aspects of the requirement understandable and not subject to misinterpretation? Is the requirement free from indefinite pronouns (this, these) and ambiguous terms (e.g., “as appropriate,” “etc.,” “and/or,” “but not limited to”)?)
- ☐ Are the requirements concise and simple?
- ☐ Do the requirements express only one thought per requirement statement, a stand-alone statement as opposed to multiple requirements in a single statement, or a paragraph that contains both requirements and rationale?
- ☐ Does the requirement statement have one subject and one predicate?

Completeness

- ☐ Are requirements stated as completely as possible? Have all incomplete requirements been captured as TBDs or TBRs and a complete listing of them maintained with the requirements?
- ☐ Are any requirements missing? For example, have any of the following requirements areas been overlooked: functional, performance, interface, environment (development, manufacturing, test, transport, storage, and operations), facility (manufacturing, test, storage, and operations), transportation (among areas for manufacturing, assembling, delivery points, within storage facilities, loading), training, personnel, operability, safety, security, appearance and physical characteristics, and design.
- ☐ Have all assumptions been explicitly stated?

Compliance

- ☐ Are all requirements at the correct level (e.g., system, segment, element, subsystem)?
- ☐ Are requirements free of implementation specifics? (Requirements should state what is needed, not how to provide it.)
- ☐ Are requirements free of descriptions of operations? (Don’t mix operation with requirements: update the ConOps instead.)
- ☐ Are requirements free of personnel or task assignments? (Don’t mix personnel/task with product requirements: update the SOW or Task Order instead.)

Consistency

- ☐ Are the requirements stated consistently without contradicting themselves or the requirements of related systems?
- ☐ Is the terminology consistent with the user and sponsor's terminology? With the project glossary?
- ☐ Is the terminology consistently used throughout the document? Are the key terms included in the project's glossary?

Traceability

- ☐ Are all requirements needed? Is each requirement necessary to meet the parent requirement? Is each requirement a needed function or characteristic? Distinguish between needs and wants. If it is not necessary, it is not a requirement. Ask, "What is the worst that could happen if the requirement was not included?"
- ☐ Are all requirements (functions, structures, and constraints) bidirectionally traceable to higher-level requirements or mission or system-of-interest scope (i.e., need(s), goals, objectives, constraints, or concept of operations)?
- ☐ Is each requirement stated in such a manner that it can be uniquely referenced (e.g., each requirement is uniquely numbered) in subordinate documents?

Correctness

- ☐ Is each requirement correct?
- ☐ Is each stated assumption correct? Assumptions should be confirmed before the document can be baselined.
- ☐ Are the requirements technically feasible?

Functionality

- ☐ Are all described functions necessary and together sufficient to meet mission and system goals and objectives?

Performance

- ☐ Are all required performance specifications and margins listed (e.g., consider timing, throughput, storage size, latency, accuracy and precision)?
- ☐ Is each performance requirement realistic?
- ☐ Are the tolerances overly tight? Are the tolerances defendable and cost-effective? Ask, "What is the worst thing that could happen if the tolerance was doubled or tripled?"

Interfaces

- ☐ Are all external interfaces clearly defined?
- ☐ Are all internal interfaces clearly defined?
- ☐ Are all interfaces necessary, sufficient, and consistent with each other?

Maintainability

- ☐ Have the requirements for maintainability of the system been specified in a measurable, verifiable manner?
- ☐ Are requirements written so that ripple effects from changes are minimized (i.e., requirements are as weakly coupled as possible)?

Reliability

- ☐ Are clearly defined, measurable, and verifiable reliability requirements specified?
- ☐ Are there error detection, reporting, handling, and recovery requirements?
- ☐ Are undesired events (e.g., single-event upset, data loss or scrambling, operator error) considered and their required responses specified?
- ☐ Have assumptions about the intended sequence of functions been stated? Are these sequences required?
- ☐ Do these requirements adequately address the survivability after a software or hardware fault of the system from the point of view of hardware, software, operations, personnel and procedures?

Verifiability/Testability

- ☐ Can the system be tested, demonstrated, inspected, or analyzed to show that it satisfies requirements? Can this be done at the level of the system at which the requirement is stated? Does a means exist to measure the accomplishment of the requirement and verify compliance? Can the criteria for verification be stated?
- ☐ Are the requirements stated precisely to facilitate specification of system test success criteria and requirements?
- ☐ Are the requirements free of unverifiable terms (e.g., flexible, easy, sufficient, safe, ad hoc, adequate, accommodate, user-friendly, usable, when required, if required, appropriate, fast, portable, light-weight, small, large, maximize, minimize, sufficient, robust, quickly, easily, clearly, other "ly" words, other "ize" words)?

Data Usage

- ☐ Where applicable, are "don't care" conditions truly "don't care"? ("Don't care" values identify cases when the value of a condition or flag is irrelevant, even though the value may be important for other cases.) Are "don't care" conditions values explicitly stated? (Correct identification of "don't care" values may improve a design's portability.)

Appendix D: Requirements Verification Matrix

When developing requirements, it is important to identify an approach for verifying the requirements. This appendix provides an example matrix that defines how all the requirements are verified. Only “shall” requirements should be included in these matrices. The matrix should identify each “shall” by unique identifier and be definitive as to the source, i.e., document from which the requirement is taken. This matrix could be divided into multiple matrices (e.g., one for each requirements document) to delineate sources of requirements depending on the project. The example is shown to provide suggested guidelines for the minimum information that should be included in the verification matrix.

Note: See appendix I for an outline of the Verification and Validation Plan. The matrix shown here (table D-1) is appendix C in that outline.

Require-ment No.	Document	Paragraph	Shall Statement	Verification Success Criteria	Verification Method	Facility or Lab	Phase[a]	Acceptanc e Require-ment?	Preflight Accept-ance?	Performing Organizatio n	Results
Unique identifier or each requirement	*Document number the requirement is contained within*	*Paragraph number of the requirement*	*Text (within reason) of the requirement, i.e., the "shall"*	*Success criteria for the requirement*	*Verification method for the requirement (analysis, inspection, demonstratio n, test)*	*Facility or laboratory used to perform the verification and validation.*	*Phase in which the verification and validation will be performed*	*Indicate whether this requirement is also verified during initial acceptance testing of each unit.*	*. Indicate whether this requiremen t is also verified during any pre-flight or recurring acceptance testing of each unit*	*Organization responsible for performing the verification*	*Indicate docu-ments that contain the objective evidence that require -ment was satisfied*
P-1	xxx	3.2.1.1 Capability: Support Uplinked Data (LDR)	System X shall provide a max. ground-to-station uplink of…	1. System X locks to forward link at the min and max data rate tolerances 2. System X locks to the forward link at the min and max operating frequency tolerances	Test	xxx	5	Yes	No	xxx	TPS xxxx
P-i	xxx	Other paragraphs	Other "shalls" in PTRS	Other criteria	xxx	xxx	xxx	Yes/No	Yes/No	xxx	Memo xxx
S-i or other unique designator	xxxxx (other specs, ICDs, etc.)	Other paragraphs	Other "shalls" in specs, ICDs, etc.	Other criteria	xxx	xxx	xxx	Yes/No	Yes/No	xxx	Report xxx

[a]. Phases defined as: (1) Pre-Declared Development, (2) Formal Box-Level Functional, (3) Formal Box-Level Environmental, (4) Formal System-Level Environmental, (5) Formal System-Level Functional, (6) Formal End-to-End Functional, (7) Integrated Vehicle Functional, (8) On-Orbit Functional.

Appendix E: Creating the Validation Plan with a Validation Requirements Matrix

Note: See appendix I for an outline of the Verification and Validation Plan. The matrix shown here (table E-1) is appendix D in that outline.

When developing requirements, it is important to identify a validation approach for how additional validation evaluation, testing, analysis, or other demonstrations will be performed to ensure customer/sponsor satisfaction.

There are a number of sources to draw from for creating the validation plan:

- ConOps
- Stakeholder/customer needs, goals, and objectives documentation
- Rationale statements for requirements and in verification requirements
- Lessons learned database
- System architecture modeling
- Test-as-you-fly design goals and constraints
- SEMP, HSIP, V&V plans

Validation products can take the form of a wide range of deliverables, including:

- Stakeholder evaluation and feedback
- Peer reviews
- Physical models of all fidelities
- Simulations
- Virtual modeling
- Tests
- Fit-checks
- Procedure dry-runs
- Integration activities (to inform on-orbit maintenance procedures)
- Phase-level review solicitation and feedback

Particular attention should be paid to the planning for life cycle phase since early validation can have a profound impact on the design and cost in the later life-cycle phases.

Table E-1 shows an example validation matrix.

Table E-1 Validation Requirements Matrix

Validation Product #	Activity	Objective	Validation Method	Facility or Lab	Phase	Performing Organization	Results
Unique identifier for validation product	*Describe evaluation by the customer/ sponsor that will be performed*	*What is to be accomplished by the customer/ sponsor evaluation*	*Validation method for the requirement (analysis, inspection, demonstration, or test)*	*Facility or laboratory used to perform the validation*	*Phase in which the verification/ validation will be performed*[a]	*Organization responsible for coordinating the validation activity*	*Indicate the objective evidence that validation activity occurred*
1	Customer/ sponsor will evaluate the candidate displays	1. Ensure legibility is acceptable 2. Ensure overall appearance is acceptable	Test	xxx	Phase A	xxx	TPS 123456

a. Example: (1) during product selection process, (2) prior to final product selection (if COTS) or prior to PDR, (3) prior to CDR, (4) during box-level functional, (5) during system-level functional, (6) during end-to-end functional, (7) during integrated vehicle functional,(8) during on-orbit functional.

Appendix F: Functional, Timing, and State Analysis

F.1 Functional Flow Block Diagrams

Functional analysis can be performed using various methods, one of which is Functional Flow Block Diagrams (FFBDs). FFBDs define the system functions and depict the time sequence of functional events. They identify “what” should happen and do not assume a particular answer to “how” a function will be performed. They are functionally oriented, not solution oriented.

FFBDs are made up of functional blocks, each of which represents a definite, finite, discrete action to be accomplished. The functional architecture is developed using a series of leveled diagrams to show the functional decomposition and display the functions in their logical, sequential relationship. A consistent numbering scheme is used to label the blocks. The numbers establish identification and relationships that carry through all the diagrams and facilitate traceability from the lower levels to the top level. Each block in the first-level (top-level) diagram can be expanded to a series of functions in the second-level diagram, and so on. (See Figure F.1-1.)

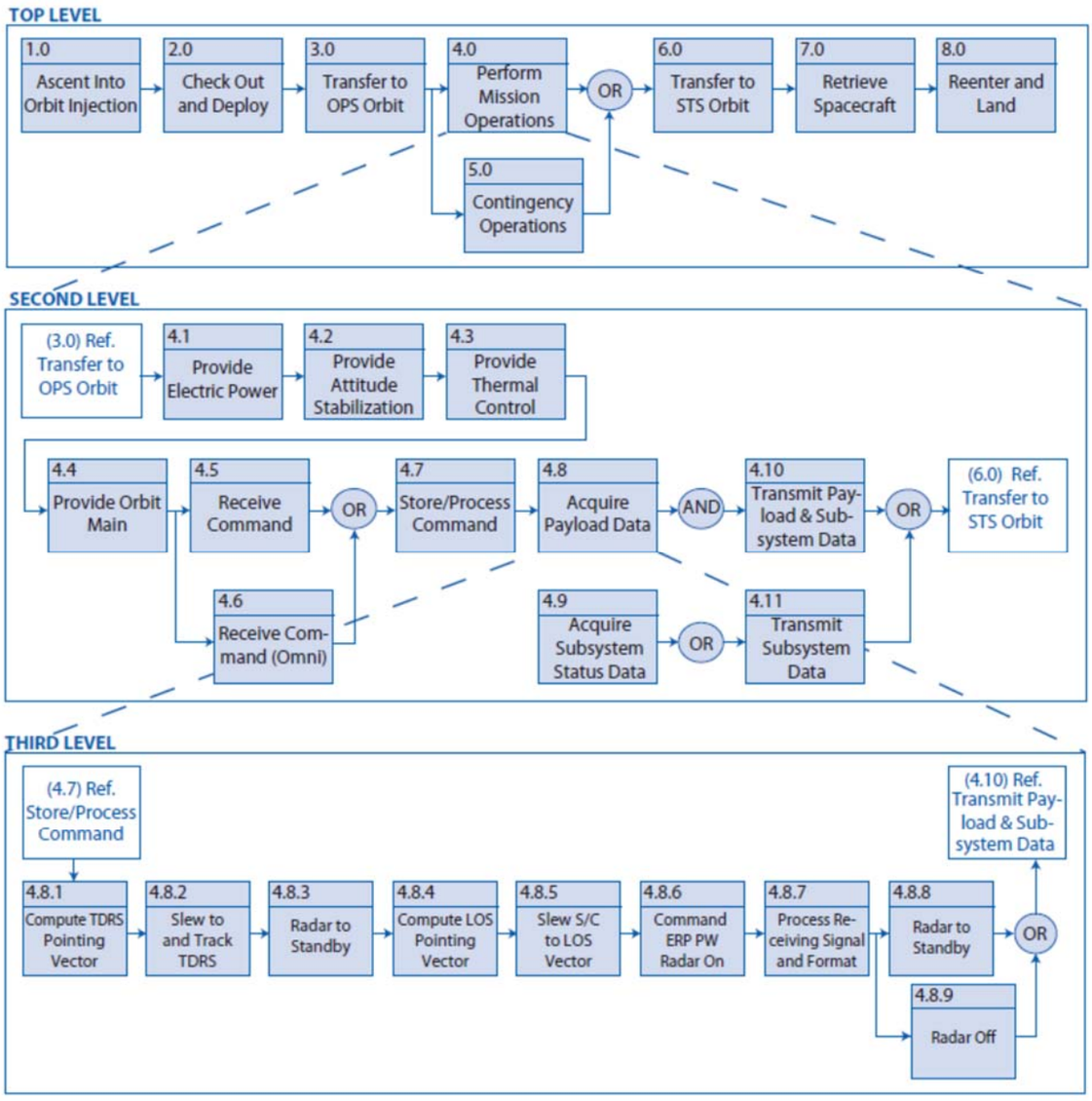


Figure F.1-1 FFBD Flowdown

Lines connecting functions indicate function flow and not lapsed time or intermediate activity. Diagrams are laid out so that the flow direction is generally from left to right. Arrows are often used to indicate functional flows. The diagrams show both input and output, thus facilitating the definition of interfaces and control process.

Each diagram contains a reference to other functional diagrams to facilitate movement between pages of the diagrams. Typically, gates are used: “AND,” “OR,” go” or “no-go,” sometimes with enhanced functionality, including exclusive OR gate (XOR), iteration (IT), repetition (RP), or loop (LP). A circle is used to denote a summing gate and is used when AND/OR is present. AND is used to indicate parallel functions and all conditions should be satisfied to proceed (i.e., concurrency). OR is used to indicate that alternative paths can be satisfied to proceed (i.e., selection). G and G ¯are used to denote “go” and “no-go” conditions, respectively. These

symbols are placed adjacent to lines leaving a particular function to indicate alternative paths. For examples of the above, see Figures F.1-2 and F.1-3.

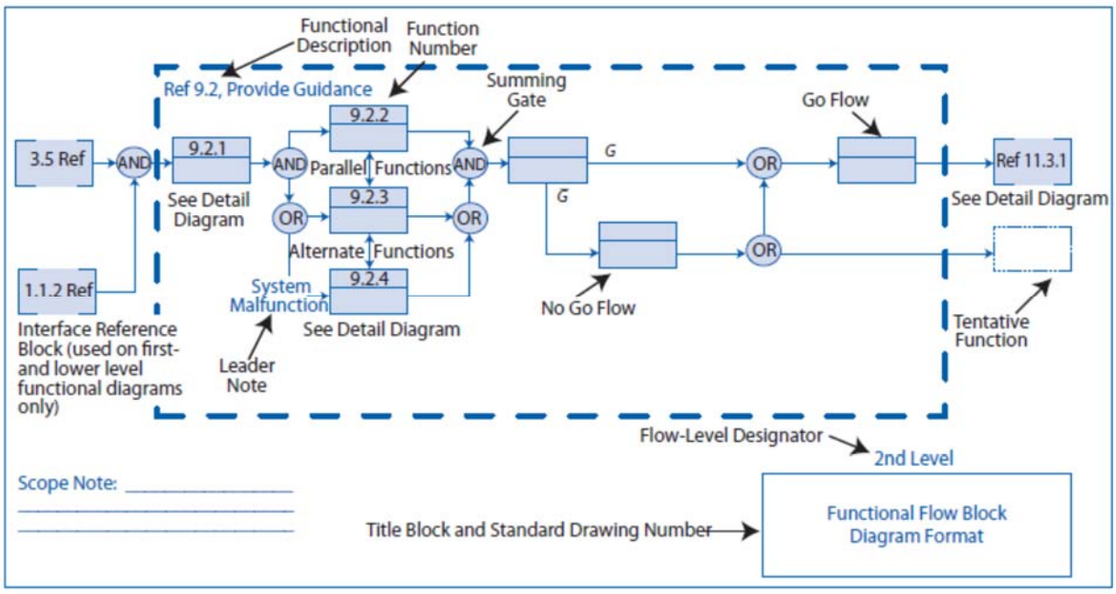


Figure F.1-2 FFBD: Example 1

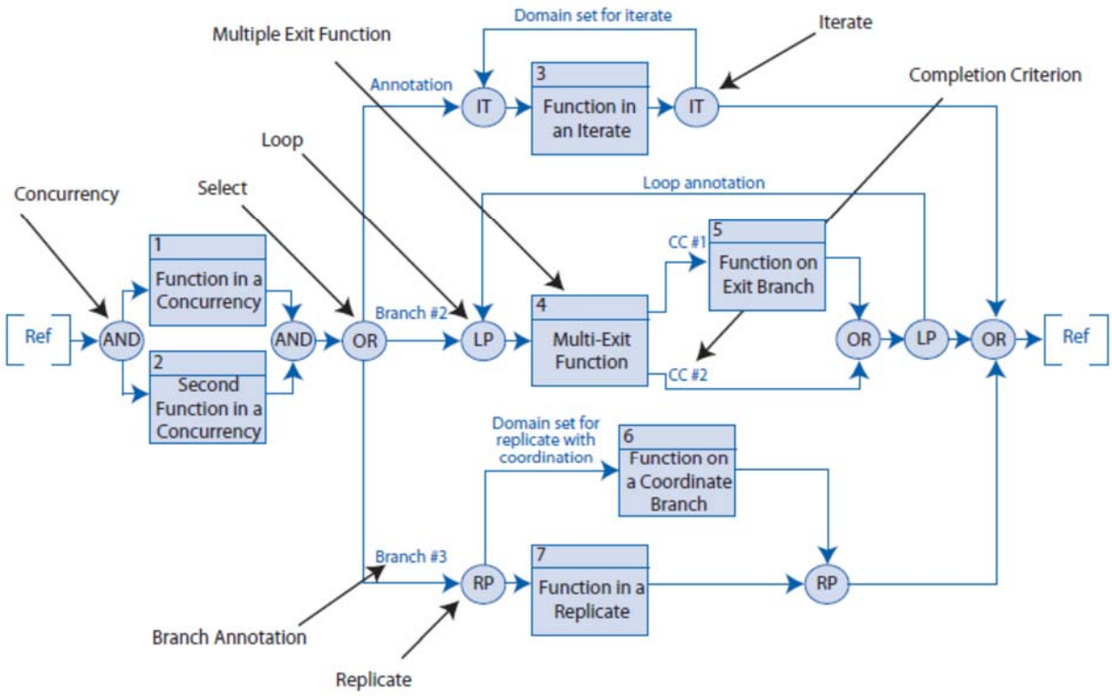


Figure F.1-3 FFBD Showing Additional Control Constructs: Example 2

Enhanced Functional Flow Block Diagrams (EFFBDs) provide data flow overlay to capture data dependencies. EFFBDs (an example is shown in Figure F.1-4) represent: (1) functions, (2) control flows, and (3) data flows. An EFFBD specification of a system is complete enough that it is executable as a discrete event model, capable of dynamic as well as static validation. EFFBDs provide freedom to use either control constructs or data triggers or both to specify execution conditions for the system functions. EFFBDs graphically differentiate triggering and non-triggering data inputs. Triggering data are required before a function can begin execution. Triggers are actually data items with control implications. In Figure F.1-4, the data input shown with double-headed arrows is a triggering data input. The non-triggering data inputs are shown with single-headed arrows. An EFFBD function is enabled by: (1) the completion of the function(s) preceding it in the control construct and (2) triggered, if trigger data are identified, before it can execute. For example, in Figure F.1-4, "1. Serial Function" should complete and "Data 3" should be present before "3. Function in Concurrency" can execute. It should be noted that the "External Input" data into "1. Serial Function" and the "External Output" data from "6. Output Function" should not be confused with the functional input and output for these functions, which are represented by the input and the output arrows respectively. Data flows are represented as elongated ovals, whereas functions are represented as rectangular boxes.

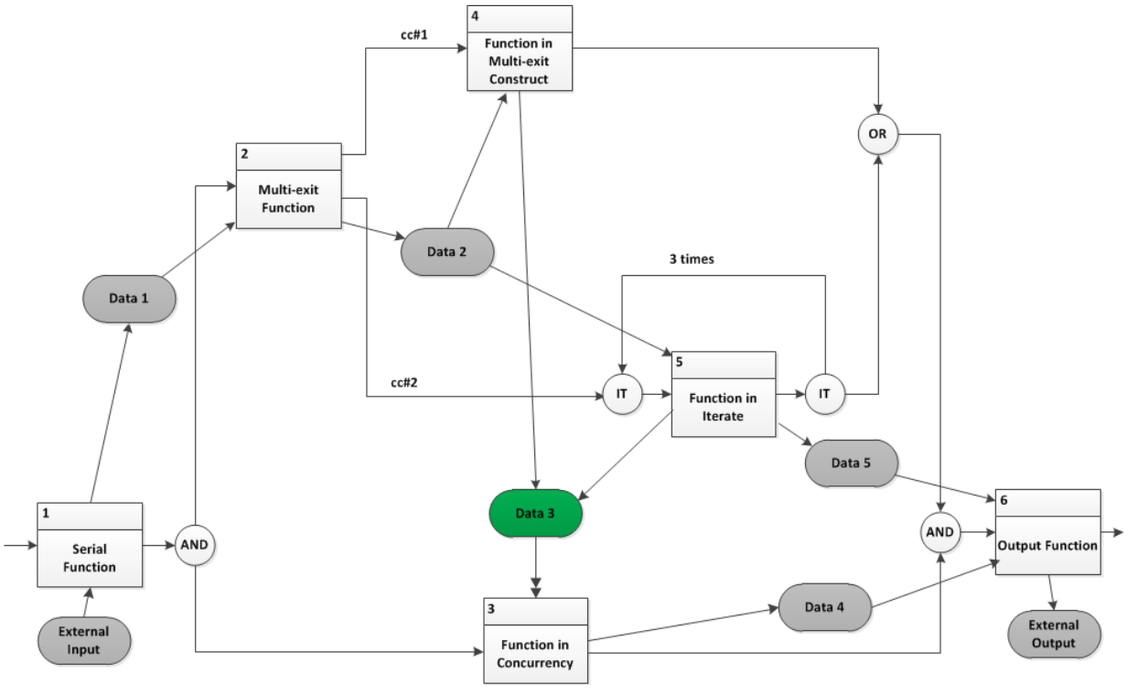


Figure F.1-4 Enhanced FFBD: Example 3

Functional analysis looks across all life cycle processes. Functions required to deploy a system are very different from functions required to operate and ultimately dispose of the system. Preparing FFBDs for each phase of the life cycle as well as the transition into the phases themselves is necessary to draw out all the requirements. These diagrams are used both to develop requirements and to identify cost-effective trade studies. The functional analysis also incorporates alternative and contingency operations, which improve the probability of mission success. The flow diagrams provide an understanding of total operation of the system, serve as a basis for development of operational and contingency procedures, and pinpoint areas where changes in operational procedures could simplify the overall system operation. In certain cases, alternative FFBDs may be used to represent various means of satisfying a particular function until data are acquired, which permits selection among the alternatives. For more information on FFBDs and EFFBDs, see Jim Long's *Relationships between Common Graphical Representations in Systems Engineering*.

F.2 Requirements Allocation Sheets / Models

Requirements allocation sheets/models document the connection between allocated functions, allocated performance, and the physical system. They provide traceability between Technical Requirements Definition functional analysis activities and Logical Decomposition and Design Solution Definition activities and maintain consistency between them, as well as show disconnects. Figure F.2-1 provides an example of a requirements allocation sheet. The reference column to the far right indicates the function numbers from the FFBDs.

Fill in the requirements allocation sheet by performing the following:

1. Include the functions and function numbers from the FFBDs.
2. Allocate functional performance requirements and design requirements to the appropriate function(s) (many requirements may be allocated to one function, or one requirement may be allocated to many functions).
3. All system-level requirements should be allocated to a function to ensure the system meets all system requirements (functions without allocated requirements should be eliminated as unnecessary activities).
4. Allocate all derived requirements to the function that spawned the requirement.
5. Identify the physical equipment, configuration item, facilities, and specifications that will be used to meet the requirements.

(For a reference on requirements allocation sheets, see DOD's *Systems Engineering Fundamentals Guide*.)

ID	DESCRIPTION	REQUIREMENT	TRACED FROM	PERFORMANCE	MARGIN	COMMENTS	REF
M1	Mission Orbit	575 +/-15 km Sun-synchronous dawn-dusk orbit	S3, S11, P3	Complies	NA	Pegasus XL with HAPS provides required launch injection dispersion accuracy	F.2.c
M2	Launch Vehicle	Pegasus XL with HAPS	P2, P4	Complies	NA		F.2.c
M3	Observatory Mass	The observatory total mass shall not exceed 241 kg	M1, M2	192.5 kg	25.20%		F.5.b
M4	Data Acquisition Quality	The mission shall deliver 95% data with better than 1 in 100,000 BER	P1	Complies	NA	Standard margins and systems baselined; formal system analysis to be completed by PDR	F.7
M5	Communication Band	The mission shall use S-band SQPSK at 5 Mbps for spacecraft downlink and 2 kbps uplink	S12, P4	Complies	NA	See SC27, SC28, and G1, G2	F.3.f, F.7
M7	Tracking	MOC shall use NORAD two-line elements for observatory tracking	P4	Complies	NA		F.7
M8	Data Latency	Data latency shall be less than 72 hours	P12	Complies	NA		F.7
M9	Daily Data Volume	Accommodate average daily raw science data volume of 10.8 Gbits	P1, S12	Complies	12%	Margin based on funded ground contacts	F.3.e, F.7
M10	Ground Station	The mission shall be compatible with the Rutherford Appleton Laboratory Ground Station and the Poker Flat Ground Station	P1	Complies	NA		F.7
M11	Orbital Debris (Casualty Area)	Design observatory for demise upon reentry with <1/10,000 probability of injury	P3	1/51,000	400%	See Orbital Debris Analysis in Appendix M-6	F.2.e, App.6
M12	Orbital Debris (Lifetime)	Design observatory for reentry <25 years after end of mission	P3	<10 years	15 years	See Orbital Debris Analysis in Appendix M-6	F.2.e, App.6

Figure F.2-1 Requirements Allocation Sheet

F.3 N2 Diagrams

An N-squared (N2) diagram is a matrix representation of functional and/or physical interfaces between elements of a system at a particular hierarchical level. The N2 diagram has been used extensively to develop data interfaces for hardware, software, and human systems. Figure F.3-1 shows an example of an N2 diagram for a hardware system. The system components are placed on the diagonal. The remaining squares in the NxN matrix represent the interfaces. The square at the intersection of a row and a column contains a description of the interface between the two components represented on that row and that column. For example, the solar arrays have a mechanical interface with the structure and an electrical interface and supplied service interface with the voltage converters. Where a blank appears, there is no interface between the respective components.

The N2 diagram can be taken down into successively lower levels to the hardware, software, and human component functional levels. In addition to defining the data that should be supplied across the interface, by showing the data flows, the N2 chart pinpoints areas where conflicts could arise in interfaces and highlights input and output dependency assumptions and requirements.

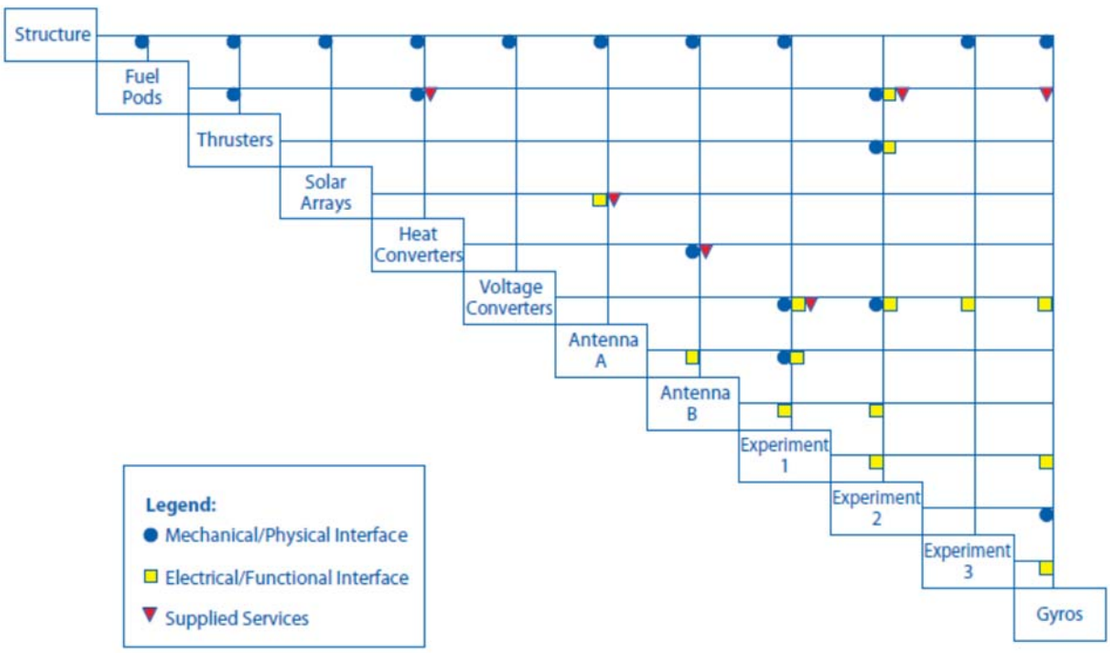


Figure F.3-1 N2 Diagram for Orbital Equipment

Source: NASA Reference Publication 1370, Training Manual for Elements of Interface Definition and Control.

F.4 Timing Analysis

There are several methods for visualizing the complex timing relationships in a system. Two of the more important ones are the timing diagram and the state transition diagram.

The timing diagram (see Figure F.4-1) defines the behavior of different objects within a timescale. It provides a visual representation of objects changing state and interacting over time. Timing diagrams can be used for defining the behavior of hardware-driven and/or software-driven and/or human-driven (crew and ground operator/maintainer) components.

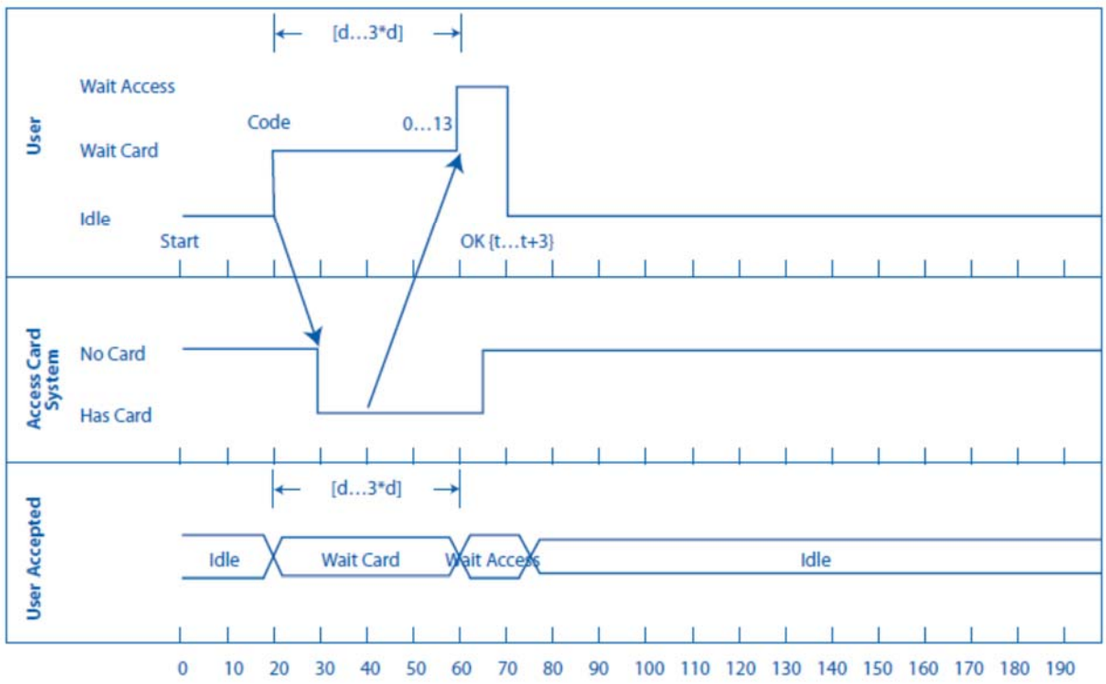


Figure F.4-1 Timing Diagram Example

Adding timing information to an FFBD to create a timeline analysis is useful for allocating resources and generating specific time-related design requirements. It also elucidates performance characteristics and design constraints. The tools of timing analysis are straightforward. While some Commercial Off-The-Shelf (COTs) tools are available, any graphics tool and a good spreadsheet can be used.

However, timing diagrams do not give a complete picture of the system. While a simple timeline analysis is useful in understanding relationships such as concurrency, overlap, and sequencing, state diagrams (see Figure F.5-1) allow for even greater flexibility in that they can depict events such as loops and decision processes that may have largely varying timelines. State diagrams are needed to show the flow of the system in response to varying inputs.

Timeline analysis is better for linear flows, while circular, looping, multi-path, and combinations of these are best described with state diagrams. Complexity should be kept layered and should track the FFBDs. The ultimate goal of using all these techniques is simply to force the thought process into the details of the system enough so that most of the big surprises can be avoided.

F.5 State Analysis

State diagramming is another graphical tool that is helpful for understanding and displaying the complex timing relationships in a system. State diagrams simplify the understanding of a system by breaking complex reactions into smaller and smaller known responses. This allows detailed requirements to be developed and verified with their timing performance.

Figure F.5-1 shows a slew command status state diagram from the James Webb Space Telescope. Ovals represent the system states. Arcs represent the event that triggers the state change as well as the action or output taken by the system in response to the event.

Self-loops are permitted. In the example in Figure F.5-1, the slew states can loop until they arrive at the correct location, and then they can loop while they settle.

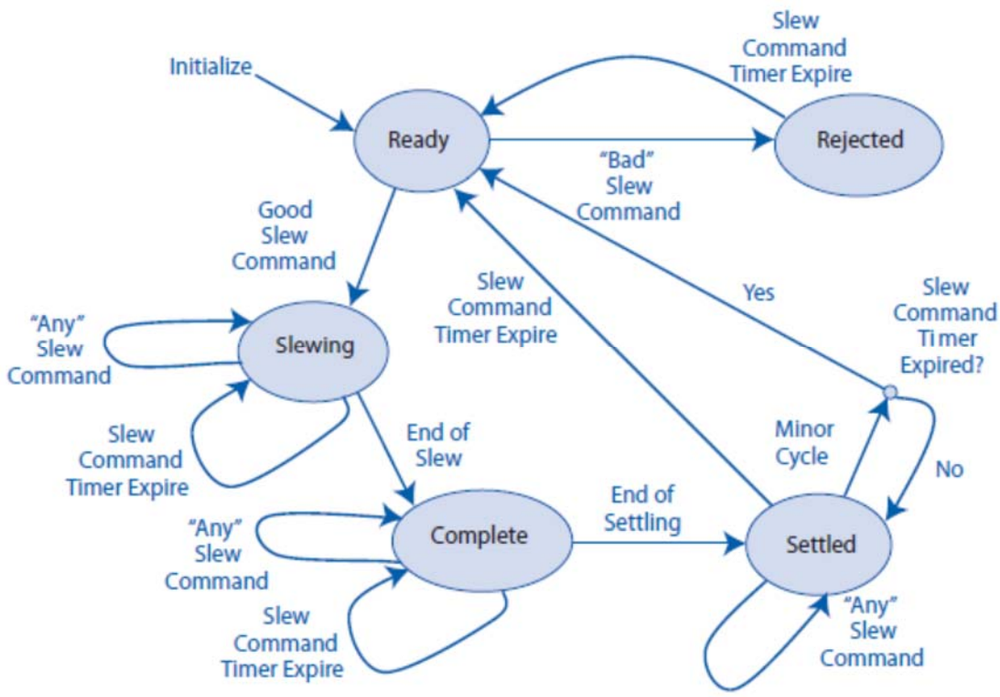
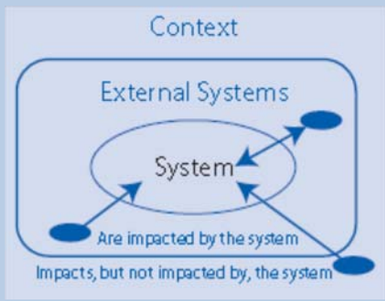


Figure F.5-1 Slew Command Status State Diagram

When it is used to represent the behavior of a sequential finite-state machine, the state diagram is called a state transition diagram. A sequential finite-state machine is one that has no memory, which means that the current output only depends on the current input. The state transition diagram models the event-based, time-dependent behavior of such a system.

Context Diagrams



When presented with a system design problem, the systems engineer's first task is to truly understand the problem. That means understanding the context in which the problem is set. A context diagram is a useful tool for grasping the system to be built and the external domains that are relevant to that system and which have interfaces to the system. The diagram shows the general structure of a context diagram. The system is shown surrounded by the external systems which have interfaces to the system. These systems are not part of the system, but they interact with the system via the system's external interfaces. The external systems can impact the system, and the system does impact the external systems. They play a major role in establishing the requirements for the system. Entities further removed are those in the system's context that can impact the system but cannot be impacted by the system. These entities in the system's context are responsible for some of the system's requirements.

Defining the boundaries of a system is a critical but often neglected task. Using an example from a satellite project, one of the external systems that is impacted by the satellite would be the Tracking and Data Relay Satellite System (TDRSS). The TDRSS is not part of the satellite system, but it defines requirements on the satellite and is impacted by the satellite since it should schedule contacts, receive and transmit data and commands, and downlink the satellite data to the ground. An example of an entity in the context of the satellite system that is not impacted by the satellite system is the Global Positioning Satellite (GPS) system. The GPS is not impacted in any way by the satellite, but it will levy some requirements on the satellite if the satellite is to use the GPS signals for navigation.

Reference: Diagram is from Buede, The Engineering Design of Systems, p. 38.

Appendix G: Technology Assessment / Insertion

G.1 Introduction, Purpose, and Scope

In 2014, the HQ Office of Chief Engineer and Office of Chief Technologist conducted an Agencywide study on Technical Readiness Level (TRL) usage and Technology Readiness Assessment (TRA) implementation. Numerous findings, observations, and recommendations were identified, as was a wealth of new guidance, best practices, and clarifications on how to interpret TRL and perform TRAs. These are presently being collected into a NASA TRA Handbook (in work), which will replace this appendix. In the interim, contact HQ/Steven Hirshorn on any specific questions on interpretation and application of TRL/TRA. Although the information contained in this appendix may change, it does provide some information until the TRA Handbook can be completed.

Agency programs and projects frequently require the development and infusion of new technological advances to meet mission goals, objectives, and resulting requirements. Sometimes the new technological advancement being infused is actually a heritage system that is being incorporated into a different architecture and operated in a different environment from that for

which it was originally designed. It is important to recognize that the adaptation of heritage systems frequently requires technological advancement. Failure to account for this requirement can result in key steps of the development process being given short shrift—often to the detriment of the program/project. In both contexts of technological advancement (new and adapted heritage), infusion is a complex process that is often dealt with in an ad hoc manner differing greatly from project to project with varying degrees of success.

Technology infusion frequently results in schedule slips, cost overruns, and occasionally even in cancellations or failures. In post mortem, the root cause of such events is often attributed to "inadequate definition of requirements." If such is indeed the root cause, then correcting the situation is simply a matter of defining better requirements, but this may not be the case—at least not totally.

In fact, there are many contributors to schedule slip, cost overrun, and project cancellation and failure—among them lack of adequate requirements definition. The case can be made that most of these contributors are related to the degree of uncertainty at the outset of the project and that a dominant factor in the degree of uncertainty is the lack of understanding of the maturity of the technology required to bring the project to fruition and a concomitant lack of understanding of the cost and schedule reserves required to advance the technology from its present state to a point where it can be qualified and successfully infused with a high degree of confidence. Although this uncertainty cannot be eliminated, it can be substantially reduced through the early application of good systems engineering practices focused on understanding the technological requirements; the maturity of the required technology; and the technological advancement required to meet program/project goals, objectives, and requirements.

A number of processes can be used to develop the appropriate level of understanding required for successful technology insertion. The intent of this appendix is to describe a systematic process that can be used as an example of how to apply standard systems engineering practices to perform a comprehensive Technology Assessment (TA). The TA comprises two parts, a Technology Maturity Assessment (TMA) and an Advancement Degree of Difficulty Assessment (AD^2). The process begins with the TMA which is used to determine technological maturity via NASA's Technology Readiness Level (TRL) scale. It then proceeds to develop an understanding of what is required to advance the level of maturity through the AD^2. It is necessary to conduct TAs at various stages throughout a program/project to provide the Key Decision Point (KDP) products required for transition between phases. (See Table G.1-1.)

Table G.1-1 Products Provided by the TA as a Function of Program/Project Phase

Gate	Product
KDP A—Transition from Pre-Phase A to Phase A	Requires an assessment of potential technology needs versus current and planned technology readiness levels, as well as potential opportunities to use commercial, academic, and other government agency sources of technology. Included as part of the draft integrated baseline. Technology Development Plan is baselined that identifies technologies to be developed, heritage systems to be modified, alternative paths to be pursued, fallback positions and corresponding performance descopes, milestones, metrics, and key decision points. Initial Technology Readiness Assessment (TRA) is available.
KDP B—Transition from Phase A to Phase B	Technology Development Plan and Technology Readiness Assessment (TRA) are updated. Incorporated in the preliminary project plan.

KDP C—Transition from Phase B to Phase C/D	Requires a TRAR demonstrating that all systems, subsystems, and components have achieved a level of technological maturity with demonstrated evidence of qualification in a relevant environment.

Source: NPR 7120.5.

The initial TMA provides the baseline maturity of the system's required technologies at program / project outset and allows monitoring progress throughout development. The final TMA is performed just prior to the Preliminary Design Review (PDR). It forms the basis for the Technology Readiness Assessment Report (TRAR), which documents the maturity of the technological advancement required by the systems, subsystems, and components demonstrated through test and analysis. The initial AD^2 provides the material necessary to develop preliminary cost and to schedule plans and preliminary risk assessments. In subsequent assessments, the information is used to build the Technology Development Plan and in the process, identify alternative paths, fallback positions, and performance descope options. The information is also vital to preparing milestones and metrics for subsequent Earned Value Management (EVM).

The TMA is performed against the hierarchical breakdown of the hardware and software products of the program/project PBS to achieve a systematic, overall understanding at the system, subsystem, and component levels. (See Figure G.1-1.)

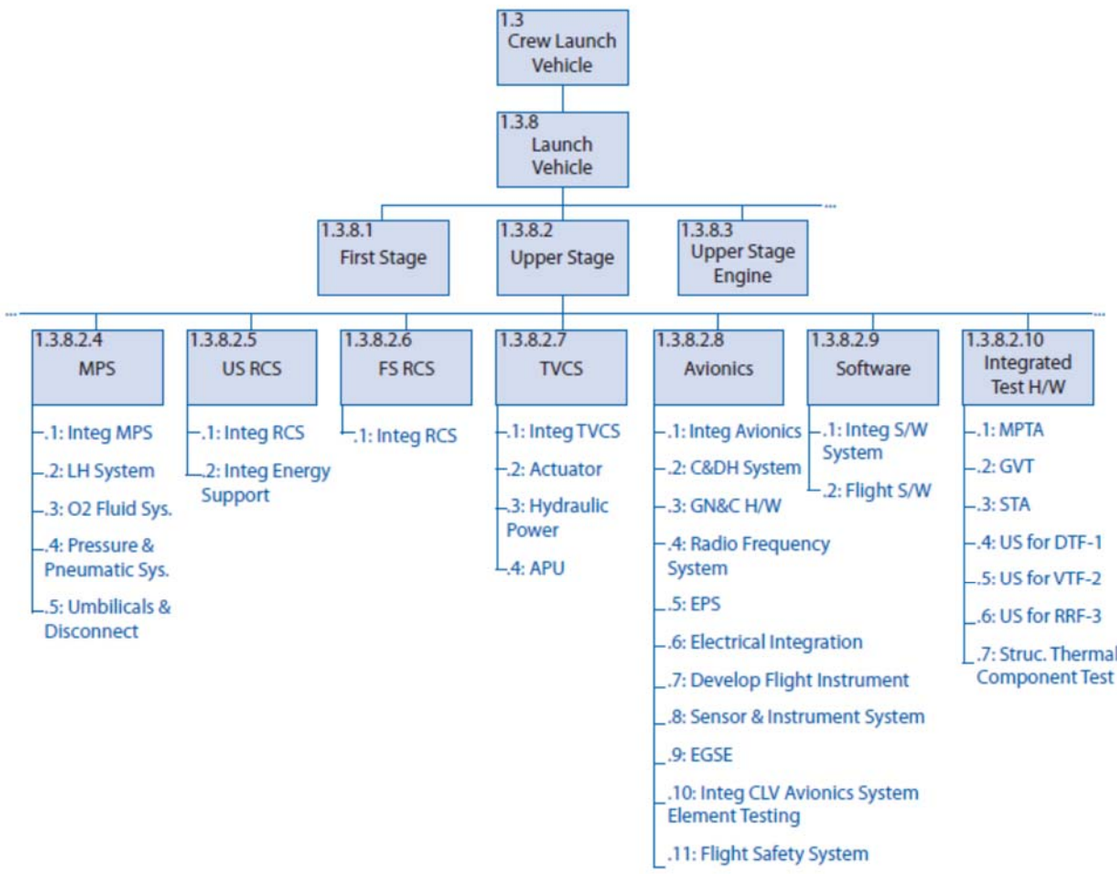


Figure G.1-1 PBS Example

G.2 Inputs / Entry Criteria

It is extremely important that a TA process be defined at the beginning of the program/project and that it be performed at the earliest possible stage (concept development) and throughout the program/project through PDR. Inputs to the process will vary in level of detail according to the phase of the program/project, and even though there is a lack of detail in Pre-Phase A, the TA will drive out the major critical technological advancements required. Therefore, at the beginning of Pre-Phase A, the following should be provided:

- Refinement of TRL definitions.
- Definition of AD^2.
- Definition of terms to be used in the assessment process.
- Establishment of meaningful evaluation criteria and metrics that will allow for clear identification of gaps and shortfalls in performance.
- Establishment of the TA team.
- Establishment of an independent TA review team.

G.3 How to Do Technology Assessment

The technology assessment process makes use of basic systems engineering principles and processes. As mentioned previously, it is structured to occur within the framework of the Product Breakdown Structure (PBS) to facilitate incorporation of the results. Using the PBS as a framework has a twofold benefit—it breaks the "problem" down into systems, subsystems, and components that can be more accurately assessed; and it provides the results of the assessment in a format that can be readily used in the generation of program costs and schedules. It can also be highly beneficial in providing milestones and metrics for progress tracking using EVM. As discussed above, it is a two-step process comprised of (1) the determination of the current technological maturity in terms of TRLs and (2) the determination of the difficulty associated with moving a technology from one TRL to the next through the use of the AD^2.

Conceptual Level Activities

The overall process is iterative, starting at the conceptual level during program Formulation, establishing the initial identification of critical technologies, and establishing the preliminary cost, schedule, and risk mitigation plans. Continuing on into Phase A, the process is used to establish the baseline maturity, the Technology Development Plan, and the associated costs and schedule. The final TA consists only of the TMA and is used to develop the TRAR, which validates that all elements are at the requisite maturity level. (See Figure G.3-1.)

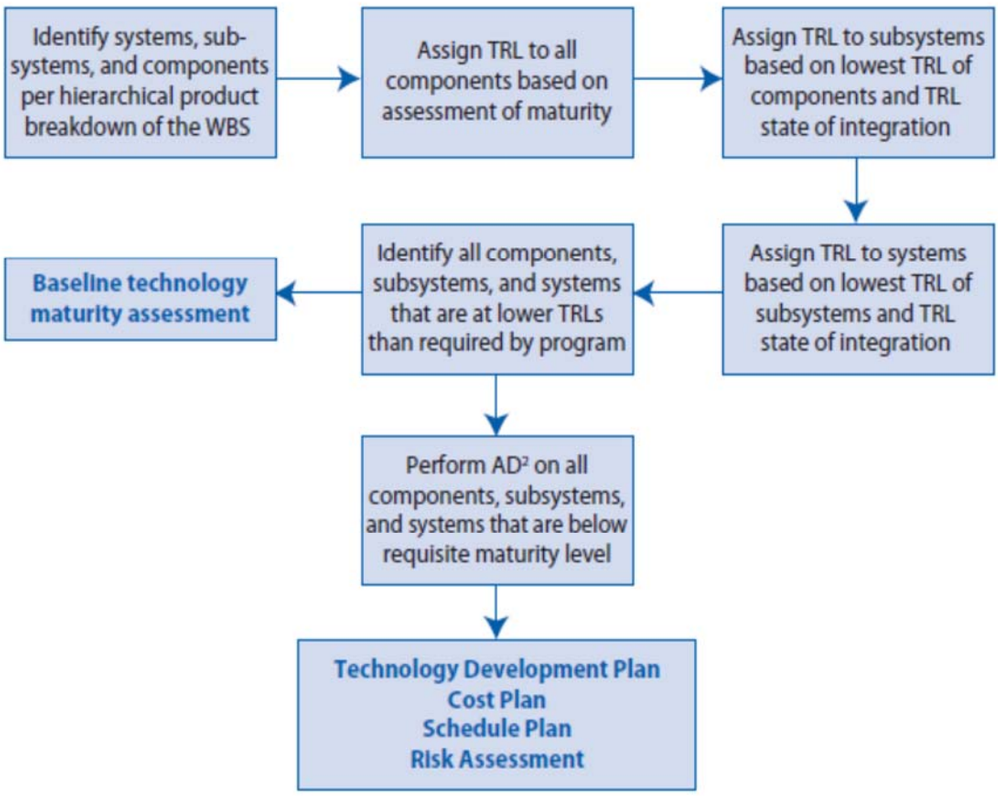


Figure G.3-1 Technology Assessment Process

Even at the conceptual level, it is important to use the formalism of a PBS to avoid allowing important technologies to slip through the cracks. Because of the preliminary nature of the concept, the systems, subsystems, and components will be defined at a level that will not permit detailed assessments to be made. The process of performing the assessment, however, is the same as that used for subsequent, more detailed steps that occur later in the program/project where systems are defined in greater detail.

Architectural Studies

Once the concept has been formulated and the initial identification of critical technologies made, it is necessary to perform detailed architecture studies with the Technology Assessment Process intimately interwoven. (See Figure G.3-2.)

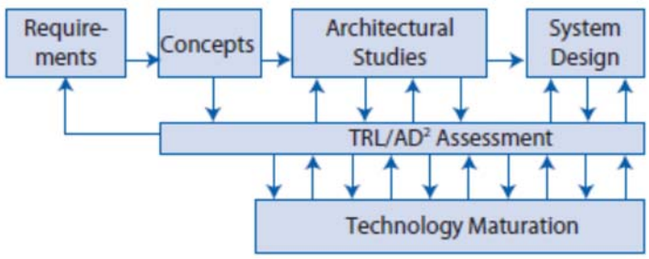


Figure G.3-2 Architectural Studies and Technology Development

The purpose of the architecture studies is to refine end-item system design to meet the overall scientific requirements of the mission. It is imperative that there be a continuous relationship between architectural studies and maturing technology advances. The architectural studies should incorporate the results of the technology maturation, planning for alternative paths and identifying new areas required for development as the architecture is refined. Similarly, it is incumbent upon the technology maturation process to identify requirements that are not feasible and development routes that are not fruitful and to transmit that information to the architecture studies in a timely manner. It is also incumbent upon the architecture studies to provide feedback to the technology development process relative to changes in requirements. Particular attention should be given to "heritage" systems in that they are often used in architectures and environments different from those in which they were designed to operate.

G.4 Establishing TRLs

A Technology Readiness Level (TRL) is, at its most basic, a description of the performance history of a given system, subsystem, or component relative to a set of levels first described at NASA HQ in the 1980s.[6] The TRL essentially describes the state of a given technology and provides a baseline from which maturity is gauged and advancement defined. (See Figure G.4-1.)

[6] The concept of a TRL was first introduced at NASA HQ by Stan Sadin in 1974 and later elaborated on by John Makins in 1995.

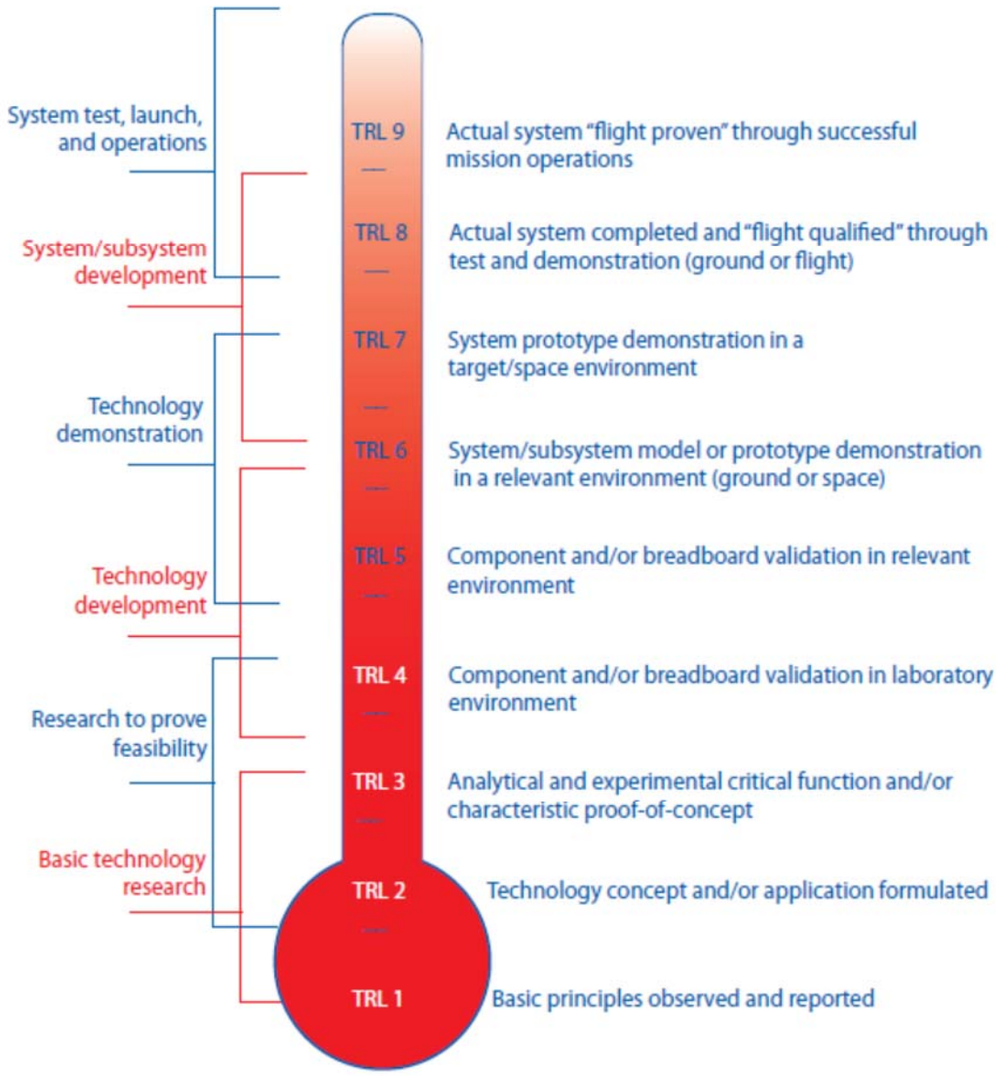


Figure G.4-1 Technology Readiness Levels

Programs are often undertaken without fully understanding either the maturity of key technologies or what is needed to develop them to the required level. *It is impossible to understand the magnitude and scope of a development program without having a clear understanding of the baseline technological maturity of all elements of the system.* Establishing the TRL is a vital first step on the way to a successful program. A frequent misconception is that in practice, it is too difficult to determine TRLs and that when you do, it is not meaningful. On the contrary, identifying TRLs can be a straightforward systems engineering process of determining what was demonstrated and under what conditions it was demonstrated.

Terminology

At first glance, the TRL descriptions in Figure G.4-1 appear to be straightforward. It is in the process of trying to assign levels that problems arise. A primary cause of difficulty is in

terminology; e.g., everyone knows what a breadboard is, but not everyone has the same definition. Also, what is a "relevant environment?" What is relevant to one application may or may not be relevant to another. Many of these terms originated in various branches of engineering and had, at the time, very specific meanings to that particular field. They have since become commonly used throughout the engineering field and often acquire differences in meaning from discipline to discipline, some differences subtle, some not so subtle. "Breadboard," for example, comes from electrical engineering where the original use referred to checking out the functional design of an electrical circuit by populating a "breadboard" with components to verify that the design operated as anticipated. Other terms come from mechanical engineering, referring primarily to units that are subjected to different levels of stress under testing, e.g., qualification, protoflight, and flight units. The first step in developing a uniform TRL assessment (see Figure G.4-2) is to define the terms used. It is extremely important to develop and use a consistent set of definitions over the course of the program/project.

Judgment Calls

Having established a common set of terminology, it is necessary to proceed to the next step: quantifying "judgment calls" on the basis of past experience. Even with clear definitions, judgment calls will be required when it comes time to assess just how similar a given element is relative to what is needed (i.e., is it close enough to a prototype to be considered a prototype, or is it more like an engineering breadboard?). Describing what has been done in terms of form, fit, and function provides a means of quantifying an element based on its design intent and subsequent performance. The current definitions for software TRLs are contained in NPR 7123.1, NASA Systems Engineering Processes and Requirements.

Assessment Team

A third critical element of any assessment relates to the question of who is in the best position to make judgment calls relative to the status of the technology in question. For this step, it is extremely important to have a well-balanced, experienced assessment team. Team members do not necessarily have to be discipline experts. The primary expertise required for a TRL assessment is that the systems engineer/user understands the current state of the art in applications. User considerations are evaluated by HFE personnel who understand the challenges of technology insertions at various stages of the product life cycle. Having established a set of definitions, defined a process for quantifying judgment calls, and assembled an expert assessment team, the process primarily consists of asking the right questions. The flowchart depicted in Figure G.4-2 demonstrates the questions to ask to determine TRL at any level in the assessment.

Heritage Systems

Note the second box particularly refers to heritage systems. If the architecture and the environment have changed, then the TRL drops to TRL 5—at least initially. Additional testing may need to be done for heritage systems for the new use or new environment. If in subsequent analysis the new environment is sufficiently close to the old environment or the new architecture sufficiently close to the old architecture, then the resulting evaluation could be TRL 6 or 7, but the most important thing to realize is that it is no longer at TRL 9. Applying this process at the system level and then proceeding to lower levels of subsystem and component identifies those elements that require development and sets the stage for the subsequent phase, determining the AD^2.

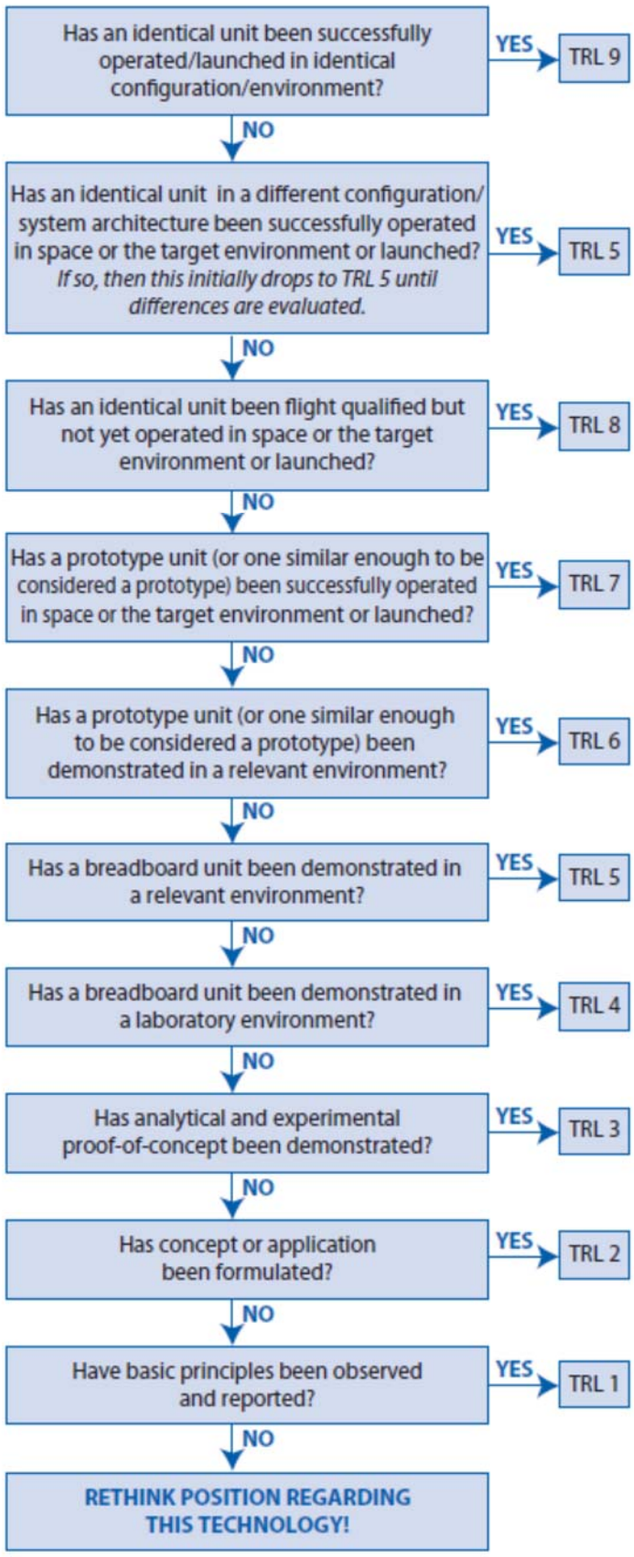


Figure G.4-2 TMA Thought Process

Formal Process for Determining TRLs

A method for formalizing this process is shown in Figure G.4-3. Here, the process has been set up as a table: the rows identify the systems, subsystems, and components that are under assessment. The columns identify the categories that will be used to determine the TRL; i.e., what units have been built, to what scale, and in what environment have they been tested. Answers to these questions determine the TRL of an item under consideration. The TRL of the system is determined by the lowest TRL present in the system; i.e., a system is at TRL 2 if any single element in the system is at TRL 2. The problem of multiple elements being at low TRLs is dealt with in the AD^2 process. Note that the issue of integration affects the TRL of every system, subsystem, and component. All of the elements can be at a higher TRL, but if they have never been integrated as a unit, the TRL will be lower for the unit. How much lower depends on the complexity of the integration. The assessed complexity depends upon the combined judgment of the engineers. It is important to have a good cross-section of senior people sitting in judgment.

TRL ASSESSMENT

	Red = Below TRL 3
	Yellow = TRL 3,4 & 5
	Green = TRL 6 and above
	White = Unknown
X	Exists

	Demonstration Units						Environment				Unit Description				
	Concept	Breadboard	Brassboard	Developmental Model	Prototype	Flight Qualified	Laboratory Environment	Relevant Environment	Space Environment	Space/Launch Operation	Form	Fit	Function	Appropriate Scale	Overall TRL
1.0 System															
1.1 Subsystem X															
1.1.1 Mechanical Components															
1.1.2 Mechanical Systems															
1.1.3 Electrical Components					X				X		X	X	X		
1.1.4 Electrical Systems															
1.1.5 Control Systems															
1.1.6 Thermal Systems							X				X	X			
1.1.7 Fluid Systems		X													
1.1.8 Optical Systems															
1.1.9 Electro-Optical Systems															
1.1.10 Software Systems															
1.1.11 Mechanisms	X														
1.1.12 Integration															
1.2 Subsystem Y															
1.2.1 Mechanical Components															

Figure G.4-3 TRL Assessment Matrix

Appendix H: Integration Plan Outline

H.1 Purpose

The integration plan defines the integration and verification strategies for a project interface with the system design and decomposition into the lower-level elements.[7] The integration plan is structured to bring the elements together to assemble each subsystem and to bring all of the subsystems together to assemble the system/product. The primary purposes of the integration plan are: (1) to describe this coordinated integration effort that supports the implementation strategy, (2) to describe for the participants what needs to be done in each integration step, and (3) to identify the required resources and when and where they will be needed.

H.2 Questions/Checklist

- Does the integration plan include and cover integration of all of the components and subsystems of the project, either developed or purchased?
- Does the integration plan account for all external systems to be integrated with the system (for example, communications networks, field equipment, other complete systems owned by the government or owned by other government agencies)?
- Does the integration plan fully support the implementation strategy, for example, when and where the subsystems and system are to be used?
- Does the integration plan mesh with the verification plan?
- For each integration step, does the integration plan define what components and subsystems are to be integrated?
- For each integration step, does the integration plan identify all the needed participants and define what their roles and responsibilities are?
- Does the integration plan establish the sequence and schedule for every integration step?
- Does the integration plan spell out how integration problems are to be documented and resolved?

H.3 Integration Plan Contents

Title Page

The title page should follow the NASA procedures or style guide. At a minimum, it should contain the following information:

- INTEGRATION PLAN FOR THE [*insert name of project]* AND [*insert name of organization]*
- *Contract number*
- *Date that the document was formally approved*

---

[7] The material in this appendix is adapted from Federal Highway Administration and CalTrans, *Systems Engineering Guidebook for ITS, Version 2.0*.

- *The organization responsible for preparing the document*
- *Internal document control number, if available*
- *Revision version and date issued*

1.0 Purpose of Document

This section gives a brief statement of the purpose of this document. It is the plan for integrating the components and subsystems of the project prior to verification.

2.0 Scope of Project

This section gives a brief description of the planned project and the purpose of the system to be built. Special emphasis is placed on the project's deployment complexities and challenges.

3.0 Integration Strategy

This section tells the reader what the high-level plan for integration is and, most importantly, why the integration plan is structured the way it is. The integration plan is subject to several, sometimes conflicting, constraints. Also, it is one part of the larger process of build, integrate, verify, and deploy, all of which should be synchronized to support the same project strategy. So, for even a moderately complex project, the integration strategy, which is based on a clear and concise statement of the project's goals and objectives, is described here at a high but all-inclusive level. It may also be necessary to describe the analysis of alternative strategies to make it clear why this particular strategy was selected.

The same strategy is the basis for the build plan, the verification plan, and the deployment plan. This section covers and describes each step in the integration process. It describes what components are integrated at each step and gives a general idea of what threads of the operational capabilities (requirements) are covered. It ties the plan to the previously identified goals and objectives so the stakeholders can understand the rationale for each integration step. This summary-level description also defines the schedule for all the integration efforts.

4.0 Phase 1 Integration

This and the following sections define and explain each step in the integration process. The intent here is to identify all the needed participants and to describe to them what they have to do. In general, the description of each integration step should identify the following:

- *The location of the activities.*
- *The project-developed equipment and software products to be integrated. Initially this is just a high-level list, but eventually the list should be exact and complete, showing part numbers and quantity.*
- *Any support equipment (special software, test hardware, software stubs, and drivers to simulate yet-to-be-integrated software components, external systems) needed for this integration step. The same support equipment is most likely needed for the subsequent verification step.*
- *All integration activities that need to be performed after installation, including integration with onsite systems and external systems at other sites.*
- *A description of the verification activities, as defined in the applicable verification plan, that occur after this integration step.*
- *The responsible parties for each activity in the integration step.*

- *The schedule for each activity.*

5.0 Multiple Phase Integration Steps (1 or N steps)

This and any needed additional sections follow the format for Section 3.0. Each covers each step in a multiple-step integration effort.

Appendix I: Verification and Validation Plan Outline

Sample Outline

The Verification and Validation (V&V) Plan needs to be baselined after the comments from PDR are incorporated. In this annotated outline, the use of the term “system” is indicative of the entire scope for which this plan is developed. This may be an entire spacecraft, just the avionics system, or a card within the avionics system. Likewise, the term “end item”, “subsystem” or “element” is meant to imply the lower-level products that, when integrated together, will produce the “system.” The general term “end item” is used to encompass activities regardless of whether the end item is a hardware or software element.

The various sections are intended to move from the high-level generic descriptions to the more detailed. The sections also flow from the lower-level items in the product layer to larger and larger assemblies and to the completely integrated system. The sections also describe how that system may be integrated and further verified/validated with its externally interfacing elements. This progression will help build a complete understanding of the overall plans for verification and validation.

1.0 Introduction

1.1 Purpose and Scope

This section states the purpose of this Verification and Validation Plan and the scope (i.e., systems) to which it applies. The purpose of the V&V Plan is to identify the activities that will establish compliance with the requirements (verification) and to establish that the system will meet the customers’ expectations (validation).

1.2 Responsibility and Change Authority

This section will identify who has responsibility for the maintenance of this plan and who or what board has the authority to approve any changes to it.

1.3 Definitions

This section will define any key terms used in the plan. The section may include the definitions of verification, validation, analysis, test, demonstration, and test. See appendix B of this guide for definitions of these and other terms that might be used.

2.0 Applicable and Reference Documents

2.1 Applicable Documents

These are the documents that may impose additional requirements or from which some of the requirements have been taken.

2.2 Reference Documents

These are the documents that are referred to within the V&V Plan that do not impose requirements, but which may have additional useful information.

2.3 Order of Precedence

This section identifies which documents take precedence whenever there are conflicting requirements.

3.0 System Description

3.1 System Requirements Flowdown

This section describes where the requirements for this system come from and how they are flowed down to subsystems and lower-level elements. It should also indicate what method will be used to perform the flowdown and bidirectional traceability of the requirements: spreadsheet, model, or other means. It can point to the file, document, or spreadsheet that captures the actual requirements flowdown.

3.2 System Architecture

This section describes the system that is within the scope of this V&V Plan. The description should be enough so that the V&V activities will have the proper context and be understandable.

3.3 End Item Architectures

This section describes each of the major end items (subsystems, elements, units, modules, etc.) that when integrated together, will form the overall system that is the scope of this V&V Plan.

3.3.1 System End Item A

This section describes the first major end item/subsystem in more detail so that the V&V activities have context and are understandable.

3.3.n System End Item n

Each end item/subsystem is separately described in a similar manner as above.

3.4 Ground Support Equipment

This section describes any major ground-support equipment that will be used during the V&V activities. This may include carts for supplying power or fuel, special test fixtures, lifting aids, simulators, or other type of support.

3.5 Other Architecture Descriptions

This section describes any other items that are important for the V&V activities but which are not included in the sections above. This may be an existing control center, training facility, or other support.

4.0 Verification and Validation Process

This section describes the process that will be used to perform verification and validation.

4.1 Verification and Validation Management Responsibilities

This section describes the responsibilities of key players in the V&V activities. It may include identification and duty description for test directors/conductors, managers, facility owners, boards, and other key stakeholders.

4.2 Verification Methods

This section defines and describes the methods that will be used during the verification activities.

4.2.1 Analysis

Defines what this verification method means (See appendix B of this guide) and how it will be applied to this system.

4.2.2 Inspection

Defines what this verification method means (See appendix B of this guide) and how it will be applied to this system.

4.2.3 Demonstration

Defines what this verification method means (See appendix B of this guide) and how it will be applied to this system.

4.2.4 Test

Defines what this verification method means (See appendix B of this guide) and how it will be applied to this system. This category may need to be broken down into further categories.

4.2.4.1 Qualification Testing

This section describes the test philosophy for the environmental and other testing that is performed at higher than normal levels to ascertain margins and performance in worst-case scenarios. Includes descriptions of how the minimum and maximum extremes will be determined for various types of tests (thermal, vibration, etc.), whether it will be performed at a component, subsystem, or system level, and the pedigree (flight unit, qualification unit, engineering unit, etc.) of the units these tests will be performed on.

4.2.4.2 Other Testing

This section describes any other testing that will be used as part of the verification activities that are not part of the qualification testing. It includes any testing of requirements within the normal operating range of the end item. It may include some engineering tests that will form the foundation or provide dry runs for the official verification testing.

4.3 Validation Methods

This section defines and describes the methods to be used during the validation activities.

4.2.1 Analysis

Defines what this validation method means (See appendix B of this guide) and how it will be applied to this system.

4.2.2 Inspection

Defines what this validation method means (See appendix B of this guide) and how it will be applied to this system.

4.2.3 Demonstration

Defines what this validation method means (See appendix B of this guide) and how it will be applied to this system.

4.2.4 Test

Defines what this validation method means (See appendix B of this guide) and how it will be applied to this system. This category may need to be broken down into further categories such as end-to-end testing, testing with humans, etc.)

4.4 Certification Process

Describes the overall process by which the results of these verification and validation activities will be used to certify that the system meets its requirements and expectations and is ready to be put into the field or fly. In addition to the verification and validation results, the certification package may also include special forms, reports, safety documentation, drawings, waivers, or other supporting documentation.

4.5 Acceptance Testing

Describes the philosophy of how/which of the verification/validation activities will be performed on each of the operational units as they are manufactured/coded and are readied for flight/use. Includes how/if data packages will be developed and provided as part of the delivery.

5.0 Verification and Validation Implementation

5.1 System Design and Verification and Validation Flow

This section describes how the system units/modules will flow from manufacturing/coding through verification and validation. Includes whether each unit will be verified/validated separately, or assembled to some level and then evaluated or other statement of flow.

5.2 Test Articles

This section describes the pedigree of test articles that will be involved in the verification/validation activities. This may include descriptions of breadboards, prototypes, engineering units, qualification units, protoflight units, flight units, or other specially named units. A definition of what is meant by these terms needs to be included to ensure clear understanding of the expected pedigree of each type of test article. Descriptions of what kind of test/analysis activities will be performed on each type of test article is included.

5.3 Support Equipment

This section describes any special support equipment that will be needed to perform the verification/validation activities. This will be a more detailed description than is stated in section 3.4 of this outline.

5.4 Facilities

This section identifies and describes major facilities that will be needed in order to accomplish the verification and validation activities. These may include environmental test facilities, computational facilities, simulation facilities, training facilities, test stands, and other facilities as needed.

6.0 End Item Verification and Validation

This section describes in detail the V&V activities that will be applied to the lower-level subsystems/elements/end items. It can point to other stand-alone descriptions of these tests if they will be generated as part of organizational responsibilities for the products at each product layer.

6.1 End Item A

This section focuses in on one of the lower-level end items and describes in detail what type of verification activities it will undergo.

6.1.1 Developmental/Engineering Unit Evaluations

This section describes what kind of testing, analysis, demonstrations, or inspections the prototype/engineering or other types of units/modules will undergo prior to performing official verification and validation.

6.1.2 Verification Activities

This section describes in detail the verification activities that will be performed on this end item.

6.1.2.1 Verification by Testing

This section describes all verification testing that will be performed on this end item.

6.1.2.1.1 Qualification Testing

This section describes the test environmental and other testing that is performed at higher than normal levels to ascertain margins and performance in worst-case scenarios. It includes what minimum and maximum extremes will be used on qualification tests (thermal, vibration, etc.) of this unit, whether it will be performed at a component, subsystem, or system level, and the pedigree (flight unit, qualification unit, engineering unit, etc.) of the units these tests will be performed on.

6.1.2.1.2 Other Testing

This section describes all other verification tests that are not performed as part of the qualification testing. These will include verification of requirements in the normal operating ranges.

6.1.2.2 Verification by Analysis

This section describes the verifications that will be performed by analysis (including verification by similarity). This may include thermal analysis, stress analysis, analysis of fracture control, materials analysis, Electrical, Electronic, and Electromechnical (EEE) parts analysis, and other analyses as needed for the verification of this end item.

6.1.2.3 Verification by Inspection

This section describes the verifications that will be performed for this end item by inspection.

6.1.2.4 Verification Demonstration

This section describes the verifications that will be performed for this end item by demonstration.

6.1.3 Validation Activities

6.1.3.1 Validation by Testing

This section describes what validation tests will be performed on this end item.

6.1.3.2 Validation by Analysis

This section describes the validation that will be performed for this end item through analysis.

6.1.3.3 Validation by Inspection

This section describes the validation that will be performed for this end item through inspection.

6.1.3.4 Validation by Demonstration

This section describes the validations that will be performed for this end item by demonstration.

6.1.4 Acceptance Testing

This section describes the set of tests, analysis, demonstrations, or inspections that will be performed on the flight/final version of the end item to show it has the same design as the one that is being verified, that the workmanship on this end item is good, and that it performs the identified functions properly.

6.n End Item n

In a similar manner as above, a description of how each end item that makes up the system will be verified and validated is made.

7.0 System Verification and Validation

7.1 End-Item Integration

This section describes how the various end items will be assembled/integrated together, verified and validated. For example, the avionics and power systems may be integrated and tested together to ensure their interfaces and performance is as required and expected prior to integration with a larger element. This section describes the verification and validation that will be performed on these major assemblies. Complete system integration will be described in later sections.

7.1.1 Developmental/Engineering Unit Evaluations

This section describes the unofficial (not the formal verification/validation) testing / analysis that will be performed on the various assemblies that will be tested together and the pedigree of the units that will be used. This may include system-level testing of configurations using engineering units, breadboard, simulators, or other forms or combination of forms.

7.1.2 Verification Activities

This section describes the verification activities that will be performed on the various assemblies.

7.1.2.1 Verification by Testing

This section describes all verification testing that will be performed on the various assemblies. The section may be broken up to describe qualification testing performed on the various assemblies and other types of testing.

7.1.2.2 Verification by Analysis

This section describes all verification analysis that will be performed on the various assemblies.

7.1.2.3 Verification by Inspection

This section describes all verification inspections that will be performed on the various assemblies.

7.1.2.4 Verification by Demonstration

This section describes all verification demonstrations that will be performed on the various assemblies.

7.1.3 Validation Activities

7.1.3.1 Validation by Testing

This section describes all validation testing that will be performed on the various assemblies.

7.1.3.2 Validation by Analysis

This section describes all validation analysis that will be performed on the various assemblies.

7.1.3.3 Validation by Inspection

This section describes all validation inspections that will be performed on the various assemblies.

7.1.3.4 Validation by Demonstration

This section describes all validation demonstrations that will be performed on the various assemblies.

7.2 Complete System Integration

This section describes the verification and validation activities that will be performed on the systems after all its assemblies are integrated together to form the complete integrated system. In some cases this will not be practical. Rationale for what cannot be done should be captured.

7.2.1 Developmental/Engineering Unit Evaluations

This section describes the unofficial (not the formal verification/validation) testing / analysis that will be performed on the complete integrated system and the pedigree of the units that will be used. This may include system-level testing of configurations using engineering units, breadboard, simulators, or other forms or combination of forms.

7.2.2 Verification Activities

This section describes the verification activities that will be performed on the completely integrated system

7.2.2.1 Verification Testing

This section describes all verification testing that will be performed on the integrated system. The section may be broken up to describe qualification testing performed at the integrated system level and other types of testing.

7.2.2.2 Verification Analysis

This section describes all verification analysis that will be performed on the integrated system.

7.2.2.3 Verification Inspection

This section describes all verification inspections that will be performed on the integrated system.

7.2.2.4 Verification Demonstration

This section describes all verification demonstrations that will be performed on the integrated system.

7.2.3 Validation Activities

This section describes the validation activities that will be performed on the completely integrated system.

7.2.3.1 Validation by Testing

This section describes all validation testing that will be performed on the integrated system.

7.2.3.2 Validation by Analysis

This section describes all validation analysis that will be performed on the integrated system.

7.2.3.3 Validation by Inspection

This section describes the validation inspections that will be performed on the integrated system.

7.2.3.4 Validation by Demonstration

This section describes the validation demonstrations that will be performed on the integrated system.

8.0 Program Verification and Validation

This section describes any further testing that the system will be subjected to. For example, if the system is an instrument, the section may include any verification/validation that the system will undergo when integrated into its spacecraft/platform. If the system is a spacecraft, the section may include any verification/validation the system will undergo when integrated with its launch vehicle.

8.1 Vehicle Integration

This section describes any further verification or validation activities that will occur when the system is integrated with its external interfaces.

8.2 End-to-End Integration

This section describes any end-to-end testing that the system may undergo. For example, this configuration would include data being sent from a ground control center through one or more relay satellites to the system and back.

8.3 On-Orbit V&V Activities

This section describes any remaining verification/validation activities that will be performed on a system after it reaches orbit or is placed in the field.

9.0 System Certification Products

This section describes the type of products that will be generated and provided as part of the certification process. This package may include the verification and validation matrix and results, pressure vessel certifications, special forms, materials certifications, test reports or other products as is appropriate for the system being verified and validated.

Appendix A: Acronyms and Abbreviations

This is a list of all the acronyms and abbreviations used in the V&V Plan and their spelled-out meaning.

Appendix B: Definition of Terms

This section is a definition of the key terms that are used in the V&V Plan.

Appendix C: Requirement Verification Matrix

The V&V Plan needs to be baselined after the comments from PDR are incorporated. The information in this section may take various forms. It could be a pointer to another document or model where the matrix and its results may be found. This works well for large projects using a requirements tracking application. The information in this section could also be the requirements matrix filled out with all but the results information and a pointer to where the results can be found. This allows the key information to be available at the time of baselining. For a smaller project, this may be the completed verification matrix. In this case, the V&V Plan would be filled out as much as possible before. See appendix D for an example of a verification matrix.

Appendix D: Validation Matrix

As with the verification matrix, this product may take various forms from a completed matrix to just a pointer for where the information can be found. Appendix E provides an example of a validation matrix.

Appendix J: SEMP Content Outline

J.1 SEMP Content

The Systems Engineering Management Plan (SEMP) is the foundation document for the technical and engineering activities conducted during the project. The SEMP conveys information to all of the personnel on the technical integration methodologies and activities for the project within the scope of the project plan. SEMP content can exist as a stand-alone document or, for smaller projects, in higher-level project documentation.

The SEMP provides the specifics of the technical effort and describes what technical processes will be used, how the processes will be applied using appropriate activities, how the project will be organized to accomplish the activities, and the resources required for accomplishing the activities. The SEMP provides the framework for realizing the appropriate work products that meet the entry and success criteria of the applicable project life-cycle phases to provide management with necessary information for assessing technical progress.

Because the SEMP provides the specific technical and management information to understand the technical integration and interfaces, its documentation and approval serve as an agreement within the project of how the technical work will be conducted. The SEMP communicates to the team itself, managers, customers, and other stakeholders the technical effort that will be performed by the assigned technical team.

The technical team, working under the overall program/project plan, develops and updates the SEMP as necessary. The technical team works with the project manager to review the content and obtain concurrence. The SEMP includes the following three general sections:

1. Technical program planning and control, which describe the processes for planning and control of the engineering efforts for the design, development, test, and evaluation of the system.
2. Systems engineering processes, which include specific tailoring of the systems engineering process as described in the NPR, implementation procedures, trade study methodologies, tools, and models to be used.
3. Engineering specialty integration describes the integration of the technical disciplines' efforts into the systems engineering process and summarizes each technical discipline effort and cross references each of the specific and relevant plans.

The SEMP outline in this appendix is guidance to be used in preparing a stand-alone project SEMP. The level of detail in the project SEMP should be adapted based on the size of the project. For a small project, the material in the SEMP can be placed in the project plan's technical summary, and this annotated outline should be used as a topic guide.

Some additional important points on the SEMP:

- The SEMP is a living document. The initial SEMP is used to establish the technical content of the engineering work early in the Formulation Phase for each project and updated as

needed throughout the project life cycle. Table J-1 provides some high level guidance on the scope of SEMP content based on the life-cycle phase.

- Project requirements that have been tailored or significant customization of SE processes should be described in the SEMP.
- For multi-level projects, the SEMP should be consistent with higher-level SEMPs and the project plan.
- For a technical effort that is contracted, the SEMP should include details on developing requirements for source selection, monitoring performance, and transferring and integrating externally produced products to NASA.

J.2 Terms Used

Terms used in the SEMP should have the same meaning as the terms used in the NPR 7123.1, Systems Engineering Processes and Requirements.

J.3 Annotated Outline

Title Page

Systems Engineering Management Plan

(Provide a title for the candidate program/project and designate a short title or proposed acronym in parenthesis, if appropriate.)

.

.

.

_____	_____
Designated Governing Authority/Technical Authority	Date
_____	_____
Program/Project Manager	Date
_____	_____
Chief Engineer	Date
_____	_____
	Date
_____	_____
	Date

By signing this document, signatories are certifying that the content herein is acceptable as direction for engineering and technical management of this program/project and that they will ensure its implementation by those over whom they have authority.

1.0 Purpose and Scope

This section provides a brief description of the purpose, scope, and content of the SEMP.

- *Purpose: This section should highlight the intent of the SEMP to provide the basis for implementing and communicating the technical effort.*
- *Scope: The scope describes the work that encompasses the SE technical effort required to generate the work products. The plan is used by the technical team to provide personnel the information necessary to successfully accomplish the required task.*
- *Content: This section should briefly describe the organization of the document.*

2.0 Applicable Documents

This section of the SEMP lists the documents applicable to this specific project and its SEMP implementation. This section should list major standards and procedures that this technical effort for this specific project needs to follow. Examples of specific procedures to list could include procedures for hazardous material handling, crew training plans for control room operations, special instrumentation techniques, special interface documentation for vehicles, and maintenance procedures specific to the project.

3.0 Technical Summary

This section contains an executive summary describing the problem to be solved by this technical effort and the purpose, context, and products to be developed and integrated with other interfacing systems identified.

Key Questions
1. What is the problem we're trying to solve?
2. What are the influencing factors?
3. What are the critical questions?
4. What are the overall project constraints in terms of cost, schedule, and technical performance
5. How will we know when we have adequately defined the problem?
6. Who are the customers?
7. Who are the users?
8. What are the customer and user priorities?
9. What is the relationship to other projects?

3.1 System Description

This section contains a definition of the purpose of the system being developed and a brief description of the purpose of the products of the product layer of the system structure to which this SEMP applies. Each product layer includes the system end products and their subsystems and the supporting or enabling products and any other work products (plans, baselines) required for the development of the system. The description should include any interfacing systems and system products, including humans with which the system products will interact physically, cognitively, functionally, or electronically.

3.2 System Structure

This section contains an explanation of how the technical portion of the product layer (including enabling products, technical cost, and technical schedule) will be developed, how

the resulting product layers will be integrated into the project portion of the WBS, and how the overall system structure will be developed. This section contains a description of the relationship of the specification tree and the drawing tree with the products of the system structure and how the relationship and interfaces of the system end products and their life cycle-enabling products will be managed throughout the planned technical effort.

3.3 Product Integration

This section contains an explanation of how the products will be integrated and describes clear organizational responsibilities and interdependencies and whether the organizations are geographically dispersed or managed across Centers. This section should also address how products created under a diverse set of contracts are to be integrated, including roles and responsibilities. This includes identifying organizations—intra-and inter-NASA, other Government agencies, contractors, or other partners—and delineating their roles and responsibilities. (See Section 7.1 of this guide.) Product integration includes the integration of analytical products.

When components or elements will be available for integration needs to be clearly understood and identified on the schedule to establish critical schedule issues.

3.4 Planning Context

This section contains the programmatic constraints (e.g., NPR 7120.5) that affect the planning and implementation of the common technical processes to be applied in performing the technical effort. The constraints provide a linkage of the technical effort with the applicable product life-cycle phases covered by the SEMP including, as applicable, milestone decision gates, major technical reviews, key intermediate events leading to project completion, life-cycle phase, event entry and success criteria, and major baseline and other work products to be delivered to the sponsor or customer of the technical effort.

3.5 Boundary of Technical Effort

This section contains a description of the boundary of the general problem to be solved by the technical effort, including technical and project constraints that affect the planning. Specifically, it identifies what can be controlled by the technical team (inside the boundary) and what influences the technical effort and is influenced by the technical effort but not controlled by the technical team (outside the boundary). Specific attention should be given to physical, cognitive, functional, and electronic interfaces across the boundary.

A description of the boundary of the system can include the following:

- *Definition of internal and external elements/items involved in realizing the system purpose as well as the system boundaries in terms of space, time, physical, and operational.*
- *Identification of what initiates the transitions of the system to operational status and what initiates its disposal is important. General and functional descriptions of the subsystems inside the boundary.*
- *Current and established subsystem performance characteristics.*
- *Interfaces and interface characteristics.*
- *Functional interface descriptions and functional flow diagrams.*
- *Key performance interface characteristics.*
- *Current integration strategies and architecture.*
- *Documented Human System Integration Plan (HSIP)*

3.6 Cross References

This section contains cross references to appropriate nontechnical plans and critical reference material that interface with the technical effort. It contains a summary description of how the technical activities covered in other plans are accomplished as fully integrated parts of the technical effort.

4.0 Technical Effort Integration

This section describes how the various inputs to the technical effort will be integrated into a coordinated effort that meets cost, schedule, and performance objectives.

The section should describe the integration and coordination of the specialty engineering disciplines into the systems engineering process during each iteration of the processes. Where there is potential for overlap of specialty efforts, the SEMP should define the relative responsibilities and authorities of each specialty. This section should contain, as needed, the project's approach to the following:

- *Concurrent engineering*
- *The activity phasing of specialty engineering*
- *The participation of specialty disciplines*
- *The involvement of specialty disciplines,*
- *The role and responsibility of specialty disciplines,*
- *The participation of specialty disciplines in system decomposition and definition*
- *The role of specialty disciplines in verification and validation*
- *Reliability*
- *Maintainability*
- *Quality assurance*
- *Integrated logistics*
- *Human engineering*
- *Safety*
- *Producibility*
- *Survivability/vulnerability*
- *National Environmental Policy Act (NEPA) compliance*
- *Launch approval/flight readiness*

The approach for coordination of diverse technical disciplines and integration of the development tasks should be described. For example, this can include the use of multidiscipline integrated teaming approaches—e.g., an HSI team—or specialized control boards. The scope and timing of the specialty engineering tasks should be described along with how specialty engineering disciplines are represented on all technical teams and during all life-cycle phases of the project.

4.1 Responsibility and Authority

This section describes the organizing structure for the technical teams assigned to this technical effort and includes how the teams will be staffed and managed.

Key Questions
1. What organization/panel will serve as the designated governing authority for this project?

Key Questions
2. How will multidisciplinary teamwork be achieved?
3. What are the roles, responsibilities, and authorities required to perform the activities of each planned common technical process?
4. What should be the planned technical staffing by discipline and expertise level?
5. What is required for technical staff training?
6. How will the assignment of roles, responsibilities, and authorities to appropriate project stakeholders or technical teams be accomplished?
7. How are we going to structure the project to enable this problem to be solved on schedule and within cost?
8. What does systems engineering management bring to the table?

The section should provide an organization chart and denote who on the team is responsible for each activity. It should indicate the lines of authority and responsibility. It should define the resolution authority to make decisions/decision process. It should show how the engineers/engineering disciplines relate.

The systems engineering roles and responsibilities need to be addressed for the following: project office, user, Contracting Officer's Representative (COR), systems engineering, design engineering, specialty engineering, and contractor.

4.2 Contractor Integration

This section describes how the technical effort of in-house and external contractors is to be integrated with the NASA technical team efforts. The established technical agreements should be described along with how contractor progress will be monitored against the agreement, how technical work or product requirement change requests will be handled, and how deliverables will be accepted. The section specifically addresses how interfaces between the NASA technical team and the contractor will be implemented for each of the 17 common technical processes. For example, it addresses how the NASA technical team will be involved with reviewing or controlling contractor-generated design solution definition documentation or how the technical team will be involved with product verification and product validation activities.

Key deliverables for the contractor to complete their systems and those required of the contractor for other project participants need to be identified and established on the schedule.

4.3 Analytical Tools that Support Integration

This section describes the methods (such as integrated computer-aided tool sets, integrated work product databases, and technical management information systems) that will be used to support technical effort integration.

5.0 Common Technical Processes Implementation

Each of the 17 common technical processes will have a separate subsection that contains a plan for performing the required process activities as appropriately tailored. (See NPR 7123.1 for the process activities required and tailoring.) Implementation of the 17 common technical processes includes (1) the generation of the outcomes needed to satisfy the entry and success criteria of the applicable product life-cycle phase or phases identified in D.4.4.4, and (2) the necessary inputs for other technical processes. These sections contain a description of the approach, methods, and tools for:

- *Identifying and obtaining adequate human and nonhuman resources for performing the planned process, developing the work products, and providing the services of the process.*
- *Assigning responsibility and authority for performing the planned process (e.g., RACI matrix, [http://en.wikipedia.org/wiki/Responsibility_assignment_matrix]), developing the work products, and providing the services of the process.*
- *Training the technical staff performing or supporting the process, where training is identified as needed.*
- *Designating and placing designated work products of the process under appropriate levels of configuration management.*
- *Identifying and involving stakeholders of the process.*
- *Monitoring and controlling the systems engineering processes.*
- *Identifying, defining, and tracking metrics and success.*
- *Objectively evaluating adherence of the process and the work products and services of the process to the applicable requirements, objectives, and standards and addressing noncompliance.*
- *Reviewing activities, status, and results of the process with appropriate levels of management and resolving issues.*

This section should also include the project-specific description of each of the 17 processes to be used, including the specific tailoring of the requirements to the system and the project; the procedures to be used in implementing the processes; in-house documentation; trade study methodology; types of mathematical and/or simulation models to be used; and generation of specifications.

Key Questions	
1.	*What are the systems engineering processes for this project?*
2.	*What are the methods that we will apply for each systems engineering task?*
3.	*What are the tools we will use to support these methods? How will the tools be integrated?*
4.	*How will we control configuration development?*
5.	*How and when will we conduct technical reviews?*
6.	*How will we establish the need for and manage trade-off studies?*
7.	*Who has authorization for technical change control?*
8.	*How will we manage requirements? interfaces? documentation?*

6.0 Technology Insertion

This section describes the approach and methods for identifying key technologies and their associated risks and criteria for assessing and inserting technologies, including those for inserting critical technologies from technology development projects. An approach should be developed for appropriate level and timing of technology insertion. This could include alternative approaches to take advantage of new technologies to meet systems needs as well as alternative options if the technologies do not prove appropriate in result or timing. The strategy for an initial technology assessment within the scope of the project requirements should be provided to identify technology constraints for the system.

Key Questions
1. *How and when will we insert new of special technology into the project?*
2. *What is the relationship to research and development efforts? How will they support the project? How will the results be incorporated?*
3. *How will we incorporate system elements provided by others? How will these items be certified for adequacy?*
4. *What facilities are required?*
5. *When and how will these items be transitioned to be part of the configuration?*

7.0 Additional SE Functions and Activities

This section describes other areas not specifically included in previous sections but that are essential for proper planning and conduct of the overall technical effort.

7.1 System Safety

This section describes the approach and methods for conducting safety analysis and assessing the risk to operators, the system, the environment, or the public.

7.2 Engineering Methods and Tools

This section describes the methods and tools that are not included in the technology insertion sections but are needed to support the overall technical effort. It identifies those tools to be acquired and tool training requirements.

This section defines the development environment for the project, including automation, simulation, and software tools. If required, it describes the tools and facilities that need to be developed or acquired for all disciplines on the project. It describes important enabling strategies such as standardizing tools across the project, or utilizing a common input and output format to support a broad range of tools used on the project. It defines the requirements for information management systems and for using existing elements. It defines and plans for the training required to use the tools and technology across the project.

7.3 Specialty Engineering

This section describes engineering discipline and specialty requirements that apply across projects and the WBS models of the system structure. Examples of these requirement areas would include planning for safety, reliability, human factors, logistics, maintainability, quality, operability, and supportability. It includes estimates of staffing levels for these disciplines and incorporates them with the project requirements.

7.4 Technical Performance Measures

a. This section describes the TPMs that have been derived from the MOEs and MOPs for the project. The TPMs are used to define and track the technical progress of the systems engineering effort. The performance metrics need to address the minimally required TPMs as defined in NPR 7123.1. These include:

1. *Mass margins for projects involving hardware* ***[SE-62]****.*
2. *Power margins for projects that are powered* ***[SE-63]****.*
3. *Review Trends including closure of review action documentation (Request for Action, Review Item Discrepancies, and/or Action Items as established by the project) for all software and hardware projects* ***[SE-64]****.*

b. *Other performance measure that should be considered by the project include:*

- *Requirement trends (percent growth, TBD/TBR closures, number of requirement changes);*
- *Interface trends (percent ICD approval, TBD/TBR burndown, number of interface requirement changes);*
- *Verification trends (closure burndown, number of deviations/waivers approved/open);*
- *Software-unique trends (number of requirements per build/release versus plan);*
- *Problem report/discrepancy report trends (number open, number closed);*
- *Cost trends (plan, actual, UFE, EVM, NOA);*
- *Schedule trends (critical path slack/float, critical milestone dates);and*
- *Staffing trends (FTE, WYE).*

Key Questions
1. *What metrics will be used to measure technical progress?*
2. *What metrics will be used to identify process improvement opportunities?*
3. *How will we measure progress against the plans and schedules?*
4. *How often will progress be reported? By whom? To whom?*

7.5 Heritage

This section describes the heritage or legacy products that will be used in the project. It should include a discussion of which products are planned to be used, the rationale for their use, and the analysis or testing needed to assure they will perform as intended in the stated use.

7.6 Other

This section is reserved to describe any unique SE functions or activities for the project that are not covered in other sections.

8.0 Integration with the Project Plan and Technical Resource Allocation

This section describes how the technical effort will integrate with project management and defines roles and responsibilities. It addresses how technical requirements will be integrated with the project plan to determine the allocation of resources, including cost, schedule, and personnel, and how changes to the allocations will be coordinated.

Key Questions
1. *How will we assess risk? What thresholds are needed for triggering mitigation activities? How will we integrate risk management into the technical decision process?*
2. *How will we communicate across and outside of the project?*
3. *How will we record decisions?*
4. *How do we incorporate lessons learned from other projects?*

This section describes the interface between all of the technical aspects of the project and the overall project management process during the systems engineering planning activities and updates. All activities to coordinate technical efforts with the overall project are included, such as technical interactions with the external stakeholders, users, and contractors.

9.0 Compliance Matrices

Appendix H.2 in NPR 7123.1A is the basis for the compliance matrix for this section of the SEMP. The project will complete this matrix from the point of view of the project and the technical scope. Each requirement will be addressed as compliant, partially compliant, or noncompliant. Compliant requirements should indicate which process or activity addresses the compliance. For example, compliance can be accomplished by using a Center process or by using a project process as described in another section of the SEMP or by reference to another documented process. Noncompliant areas should state the rationale for noncompliance.

Appendices

Appendices are included, as necessary, to provide a glossary, acronyms and abbreviations, and information published separately for convenience in document maintenance. Included are: (1) information that may be pertinent to multiple topic areas (e.g., description of methods or procedures); (2) charts and proprietary data applicable to the technical efforts required in the SEMP; and (3) a summary of technical plans associated with the project. Each appendix should be referenced in one of the sections of the engineering plan where data would normally have been provided.

Templates

Any templates for forms, plans, or reports the technical team will need to fill out, like the format for the verification and validation plan, should be included in the appendices.

References

This section contains all documents referenced in the text of the SEMP.

Table J-1 Guidance on SEMP Content per Life-Cycle Phase

SEMP Section	SEMP Subsection	Pre-Phase A KDP A	Phase A KDP B		Phase B KDP C	Phase C KDP D		Phase D KDP E		Phase E KDP F	Phase F
		MCR	SRR	SDR/MDR	PDR	CDR	SIR	ORR	MRR/FRR	DR	DRR
Purpose and Scope		Final	Final	Final	Final	Final	Final	Final	Final	Final	Final
Applicable Documents		Initial	Initial	Initial	Final	Final	Final	Final	Final	Final	Final
Technical Summary		Final	Final	Final	Final	Final	Final	Final	Final	Final	Final
System Description		Initial	Initial	Initial	Final	Final	Final	Final	Final	Final	Final
System Structure	Product Integration	Define thru SDR timeframe	Define thru SDR timeframe	Define thru SDR timeframe	Define thru SIR	Define thru SIR	Define thru SIR	Define sustaining thru end of program	Define sustaining thru end of program	Define sustaining thru end of program	Define sustaining thru end of program
	Planning Context	Define thru SDR timeframe	Define thru SDR timeframe	Define thru SDR timeframe	Define thru SIR	Define thru SIR	Define thru SIR	Define sustaining thru end of program	Define sustaining thru end of program	Define sustaining thru end of program	Define sustaining thru end of program
	Boundary of Technical Effort	Initial	Initial	Initial	Final	Final	Final	Final	Final	Final	Final
	Cross References	Initial	Initial	Initial	Final	Final	Final	Final	Final	Final	Final
Technical Effort Integration	Responsibility and Authority	Define thru SDR timeframe	Define thru SDR timeframe	Define thru SDR timeframe	Define thru SIR timeframe	Define thru SIR timeframe	Define thru SIR timeframe	Define sustaining Roles and Responsibilities through end of program	Define sustaining Roles and Responsibilities through end of program	Define sustaining Roles and Responsibilities through end of program	
	Contractor Integration	Define acquisitions needed		Define insight/ oversight through SIR timeframe				Define sustaining insight/ oversight through end of program			
	Support Integration	Define acquisitions needed		Define insight/ oversight through SIR timeframe				Define sustaining insight/ oversight through end of program			

SEMP Section	SEMP Subsection	Pre-Phase A KDP A	Phase A KDP B		Phase B KDP C	Phase C KDP D		Phase D KDP E		Phase E KDP F	Phase F
		MCR	SRR	SDR/MDR	PDR	CDR	SIR	ORR	MRR/FRR	DR	DRR
Common Technical Processes Implementation		Processes defined for Concept Development and Formulation		Processes defined for the Design Phase		Processes added for the integration and Operations Phase		Update Operations processes. Define close out processes and sustaining engineering processes			
Technology Insertion		Define technologies to be developed		Define decision process for on ramps and off ramps of technology efforts				Define technology sustaining effort through end of program.			
Additional SE Functions and Activities	System Safety	Define process through CDR						Define sustaining Roles and Responsibilities through end of program			
	Engineering Methods and tools	Define process through CDR						Define sustaining Roles and Responsibilities through end of program			
	Specialty Engineering	Define process through CDR						Define sustaining Roles and Responsibilities through end of program			
Integration with the Project Plan and Technical Resource Allocation		Define through SDR timeframe			Define through SIR	Define through SIR	Define through SIR	Define sustaining through end of program	Define sustaining through end of program	Define sustaining through end of program	Define sustaining through end of program
Compliance Matrix		Initial	Initial	Initial	Final	Final	Final	Final	Final	Final	Final

SEMP Section	SEMP Subsection	Pre-Phase A KDP A	Phase A KDP B		Phase B KDP C	Phase C KDP D		Phase D KDP E		Phase E KDP F	Phase F
		MCR	SRR	SDR/MDR	PDR	CDR	SIR	ORR	MRR/FRR	DR	DRR
(Appendix H.2 of SE NPR)											
Appendices		As required	As required	As required	As required	As required	As required	As required	As required	As required	As required
Templates		As required	As required	As required	As required	As required	As required	As required	As required	As required	As required
References		As required	As required	As required	As required	As required	As required	As required	As required	As required	As required

Appendix K: Technical Plans

The following table represents a typical expectation of maturity of some of the key technical plans developed during the SE processes. This example is for a space flight project. Requirements for work product maturity can be found in the governing PM document (i.e., NPR 7120.5) for the associated type of project.

Table K-1 Example of Expected Maturity of Key Technical Plans

Plan	Pre-Phase A	Phase A		Phase B	Phase C		Phase D		Phase E	Phase F	Ref. Page
	MCR	SRR	SDR/ MDR	PDR	CDR	SIR	ORR	MRR/ FRR	DR	DRR	
Systems Engineering Management Plan	P	B	U	U	U	U	U	U	U	U	
Risk Management Plan	A	B	U	U	U						
Integrated Logistics Support Plan	A	P	P	B	U						
Technology Development Plan	B	U	U	U							
Review Plan	P	B	U	U	U	U	U	U	U	U	
Verification and Validation Plan	A	A	P	B	U						
Integration Plan			P	B	U						
Configuration Management Plan		B	U	U							
Data Management Plan		B	U	U							
Human Systems Integration Plan		B	U	U	U						
Software Management Plan		P	B	U							
Reliability and Maintainability Plan			P	B	U						
Mission Operations Plan						P	B	U			
Project Protection Plan			P	B	U	U	U	U	U	U	
Decommissioning Plan			A					B	U		
Disposal Plan			A					B	U	U	

A= Approach B = Baseline P = Preliminary U = Update

Appendix L: Interface Requirements Document Outline

1.0 Introduction

1.1 Purpose and Scope

State the purpose of this document and briefly identify the interface to be defined. (For example, "This IRD defines and controls the interface(s) requirements between _____ and _____.")

1.2 Precedence

Define the relationship of this document to other program documents and specify which is controlling in the event of a conflict.

1.3 Responsibility and Change Authority

State the responsibilities of the interfacing organizations for development of this document and its contents. Define document approval authority (including change approval authority).

2.0 Documents

2.1 Applicable Documents

List binding documents that are invoked to the extent specified in this IRD. The latest revision or most recent version should be listed. Documents and requirements imposed by higher-level documents (higher order of precedence) should not be repeated.

2.2 Reference Documents

List any document that is referenced in the text in this subsection.

3.0 Interfaces

3.1 General

In the subsections that follow, provide the detailed description, responsibilities, coordinate systems, and numerical requirements as they relate to the interface plane.

3.1.1 Interface Description

Describe the interface as defined in the system specification. Use tables, figures, or drawings as appropriate.

3.1.2 Interface Responsibilities

Define interface hardware and interface boundary responsibilities to depict the interface plane. Use tables, figures, or drawings as appropriate.

3.1.3 Coordinate Systems

Define the coordinate system used for interface requirements on each side of the interface. Use tables, figures, or drawings as appropriate.

3.1.4 Engineering Units, Tolerances, and Conversion.

Define the measurement units along with tolerances. If required, define the conversion between measurement systems.

3.2 Interface Requirements

In the subsections that follow, define structural limiting values at the interface, such as interface loads, forcing functions, and dynamic conditions. Define the interface requirements on each side of the interface plane.

3.2.1 Mass Properties

Define the derived interface requirements based on the allocated requirements contained in the applicable specification pertaining to that side of the interface. For example, this subsection should cover the mass of the element.

3.2.2 Structural/Mechanical

Define the derived interface requirements based on the allocated requirements contained in the applicable specification pertaining to that side of the interface. For example, this subsection should cover attachment, stiffness, latching, and mechanisms.

3.2.3 Fluid

Define the derived interface requirements based on the allocated requirements contained in the applicable specification pertaining to that side of the interface. For example, this subsection should cover fluid areas such as thermal control, O_2 and N_2, potable and waste water, fuel cell water, and atmospheric sampling.

3.2.4 Electrical (Power)

Define the derived interface requirements based on the allocated requirements contained in the applicable specification pertaining to that side of the interface. For example, this subsection should cover various electric current, voltage, wattage, and resistance levels.

3.2.5 Electronic (Signal)

Define the derived interface requirements based on the allocated requirements contained in the applicable specification pertaining to that side of the interface. For example, this subsection should cover various signal types such as audio, video, command data handling, and navigation.

3.2.6 Software and Data

Define the derived interface requirements based on the allocated requirements contained in the applicable specification pertaining to that side of the interface. For example, this subsection should cover various data standards, message timing, protocols, error detection/correction, functions, initialization, and status.

3.2.7 Environments

Define the derived interface requirements based on the allocated requirements contained in the applicable specification pertaining to that side of the interface. For example, cover the dynamic envelope measures of the element in English units or the metric equivalent on this side of the interface.

3.2.7.1 Electromagnetic Effects

3.2.7.1.a Electromagnetic Compatibility

Define the appropriate electromagnetic compatibility requirements. For example, the end-item-1-to-end-item-2 interface shall meet the requirements [to be determined] of systems requirements for electromagnetic compatibility.

3.2.7.1.b Electromagnetic Interference

Define the appropriate electromagnetic interference requirements. For example, the end-item-1-to-end-item-2 interface shall meet the requirements [to be determined] of electromagnetic emission and susceptibility requirements for electromagnetic compatibility.

3.2.7.1.c Grounding

Define the appropriate grounding requirements. For example, the end-item-1-to-end-item-2 interface shall meet the requirements [to be determined] of grounding requirements.

3.2.7.1.d Bonding

Define the appropriate bonding requirements. For example, the end-item-1-to-end-item-2 structural/mechanical interface shall meet the requirements [to be determined] of electrical bonding requirements.

3.2.7.1.e Cable and Wire Design

Define the appropriate cable and wire design requirements. For example, the end-item-1-to-end-item-2 cable and wire interface shall meet the requirements [to be determined] of cable/wire design and control requirements for electromagnetic compatibility.

3.2.7.2 Acoustic

Define the appropriate acoustics requirements. Define the acoustic noise levels on each side of the interface in accordance with program or project requirements.

3.2.7.3 Structural Loads

Define the appropriate structural loads requirements. Define the mated loads that each end item should accommodate.

3.2.7.4 Vibroacoustics

Define the appropriate vibroacoustics requirements. Define the vibroacoustic loads that each end item should accommodate.

3.2.7.5 Human Operability

Define the appropriate human interface requirements. Define the human-centered design considerations, constraints, and capabilities that each end item should accommodate.

3.2.8 Other Types of Interface Requirements

Define other types of unique interface requirements that may be applicable.

Appendix M: CM Plan Outline

A comprehensive Configuration Management (CM) Plan that reflects efficient application of configuration management principles and practices would normally include the following topics:

- General product definition and scope
- Description of CM activities and procedures for each major CM function
- Organization, roles, responsibilities, and resources
- Definitions of terms
- Programmatic and organizational interfaces
- Deliverables, milestones, and schedules
- Subcontract flow down requirements

The documented CM planning should be reevaluated following any significant change affecting the context and environment, e.g., changes in suppliers or supplier responsibilities, changes in diminishing manufacturing sources (DMS)/part obsolescence, changes in resource availabilities, changes in customer contract, and changes in the product. CM planning should also be reviewed on a periodic basis to make sure that an organization's application of CM functions is current.

For additional information regarding a CM Plan, refer to SAE EIA-649, Rev. B.

Appendix N: Guidance on Technical Peer Reviews/Inspections

N.1 Introduction

The objective of technical peer reviews/inspections is to remove defects as early as possible in the development process. Peer reviews/inspections are a well-defined review process for finding and fixing defects, conducted by a team of peers with assigned roles. Peer reviews/inspections are held within development phases or between milestone reviews on completed products or completed portions of products. The results of peer reviews/inspections can be reported at milestone reviews. Checklists are heavily utilized in peer reviews/inspections to improve the quality of the review.

Technical peer reviews/inspections have proven over time to be one of the most effective practices available for ensuring quality products and on-time deliveries. Many studies have demonstrated their benefits, both within NASA and across industry. Peer reviews/inspections improve quality and reduce cost by reducing rework. The studies have shown that the rework effort saved not only pays for the effort spent on inspections, but also provides additional cost savings on the project. By removing defects at their origin (e.g., requirements and design documents, test plans and procedures, software code, etc.), inspections prevent defects from propagating through multiple phases and work products, and reduce the overall amount of rework necessary on projects. In addition, improved team efficiency is a side effect of peer reviews/inspections; e.g., by improving team communication, more quickly bringing new members up to speed, and educating project members about effective development practices.

The following section describes an example of a formal review process. Process formality may vary between projects depending on size and complexity.

N.2 How to Perform Technical Peer Reviews / Inspections

Figure N.2-1 shows a diagram of the peer review/inspection stages, and the text below the figure explains how to perform each of the stages. (Figure N.2-2 at the end of the appendix summarizes the information as a quick reference guide.)

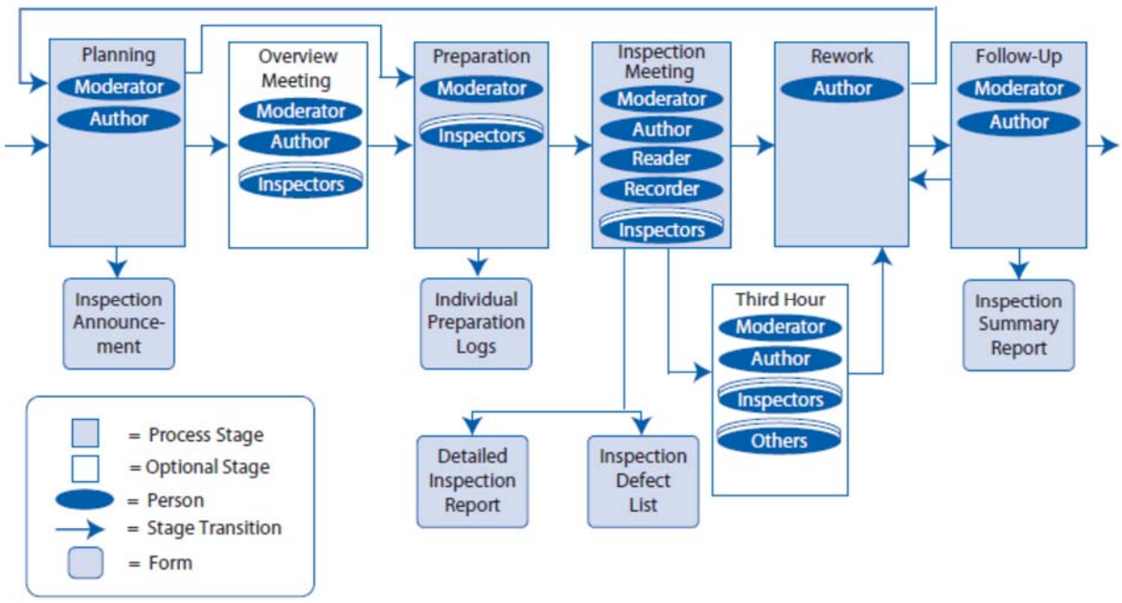


Figure N.2-1 Peer Review / Inspection Process

It is recommended that the moderator review the text blocks entitled "Planning Inspection Schedule and Estimating Staff Hours", "Guidelines for Successful Inspections", and "10 Basic Rules of Inspections" in Figure N.2-2 before beginning the planning stage. (Note: NPR 7150.2, NASA Software Engineering Requirements, defines Agency requirements on the use of peer reviews and inspections for software development. NASA peer review/inspection training is offered by the NASA Office of the Chief Engineer.)

Note: Where activities have an *, the moderator records the time on the inspection summary report.

Planning

The moderator of the peer review/inspection performs the following activities.[8]

1. Determine whether peer review/inspection entrance criteria have been met.
2. Determine whether an overview of the product is needed.
3. Select the peer review/inspection team and assign roles. For guidance on roles, see the text block entitled "Roles of Participants" in Figure N-2 at the end of this appendix. Reviewers have a vested interest in the work product; i.e., they are peers representing areas of the life cycle affected by the material being reviewed.
4. Determine if the size of the product is within the prescribed guidelines for the type of inspection. (See the text block "Meeting Rate Guidelines" in Figure N-2 for guidelines on the

---

[8] Langley Research Center, *Instructional Handbook for Formal Inspections*. This document provides more detailed instructions on how to perform technical peer reviews/inspections. It also provides templates for the forms used in the peer review/inspection process described above: inspection announcement, individual preparation log, inspection defect list, detailed inspection report, and the inspection summary report.

optimal number of pages or lines of code to inspect for each type of inspection.) If the product exceeds the prescribed guidelines, break the product into parts and inspect each part separately. (It is highly recommended that the peer review/inspection meeting not exceed 2 hours.)

5. Schedule the overview (if one is needed).
6. Schedule peer review/inspection meeting time and place.
7. Prepare and distribute the inspection announcement and package. Include in the package the product to be reviewed and the appropriate checklist for the peer review/inspection. For example, when performing a peer review on a requirements document, you can base your peer review checklist on appendix C and your Center's annotated outline for a requirements document. When performing a peer review on a SEMP, the requirements are listed in NPR 7123.1, and you can base your peer review checklist on the annotated outline for a SEMP, also in NPR 7123.1.
8. Record total time spent in planning.*

Overview Meeting

1. Moderator runs the meeting, and the author presents background information to the reviewers.
2. Record total time spent in the overview.*

Peer Review / Inspection Preparation

1. Peers review the checklist definitions of defects.
2. Examine materials for understanding and possible defects.
3. Prepare for assigned role in peer review/inspection.
4. Complete and turn in individual preparation log to the moderator.
5. The moderator reviews the individual preparation logs and makes a go or no-go decision and organizes inspection meeting.
6. Record total time spent in the preparation.*

Peer Review / Inspection Meeting

1. The moderator introduces people and identifies their peer review/inspection roles.
2. The reader presents work products to the peer review/ inspection team in a logical and orderly manner.
3. Peer reviewers/inspectors find and classify defects by severity, category, and type. (See Classification of Defects in Figure N.2-2.)
4. The recorder writes the major and minor defects on the inspection defect list. (For definitions of major and minor, see the Severity section of Figure N.2-2.)
5. Steps 1 through 4 are repeated until the review of the product is completed.
6. Open issues are assigned to peer reviewers/inspectors if irresolvable discrepancies occur.
7. Summarize the number of defects and their classification on the detailed inspection report.

8. Determine the need for a reinspection or third hour. Optional: Trivial defects (e.g., redlined documents) can be given directly to the author at the end of the inspection.
9. The moderator obtains an estimate for rework time and completion date from the author, and does the same for action items if appropriate.
10. The moderator assigns writing of change requests and/or problem reports (if needed).
11. Record total time spent in the peer review/inspection meeting.*

Third Hour

1. Complete assigned action items and provide information to the author.
2. Attend third-hour meeting at author's request.
3. Provide time spent in third hour to the moderator.*

Rework

1. All major defects noted in the inspection defect list are resolved by the author.
2. Minor and trivial defects (which would not result in faulty execution) are resolved at the discretion of the author as time and cost permit.
3. Record total time spent in the rework on the inspection defect list.

Followup

1. The moderator verifies that all major defects have been corrected and no secondary defects have been introduced.
2. The moderator ensures all open issues are resolved and verifies all success criteria for the peer review/ inspection are met.
3. Record total time spent in rework and followup.*
4. File the inspection package.
5. The inspection summary report is distributed.
6. Communicate that the peer review/inspection has been passed.

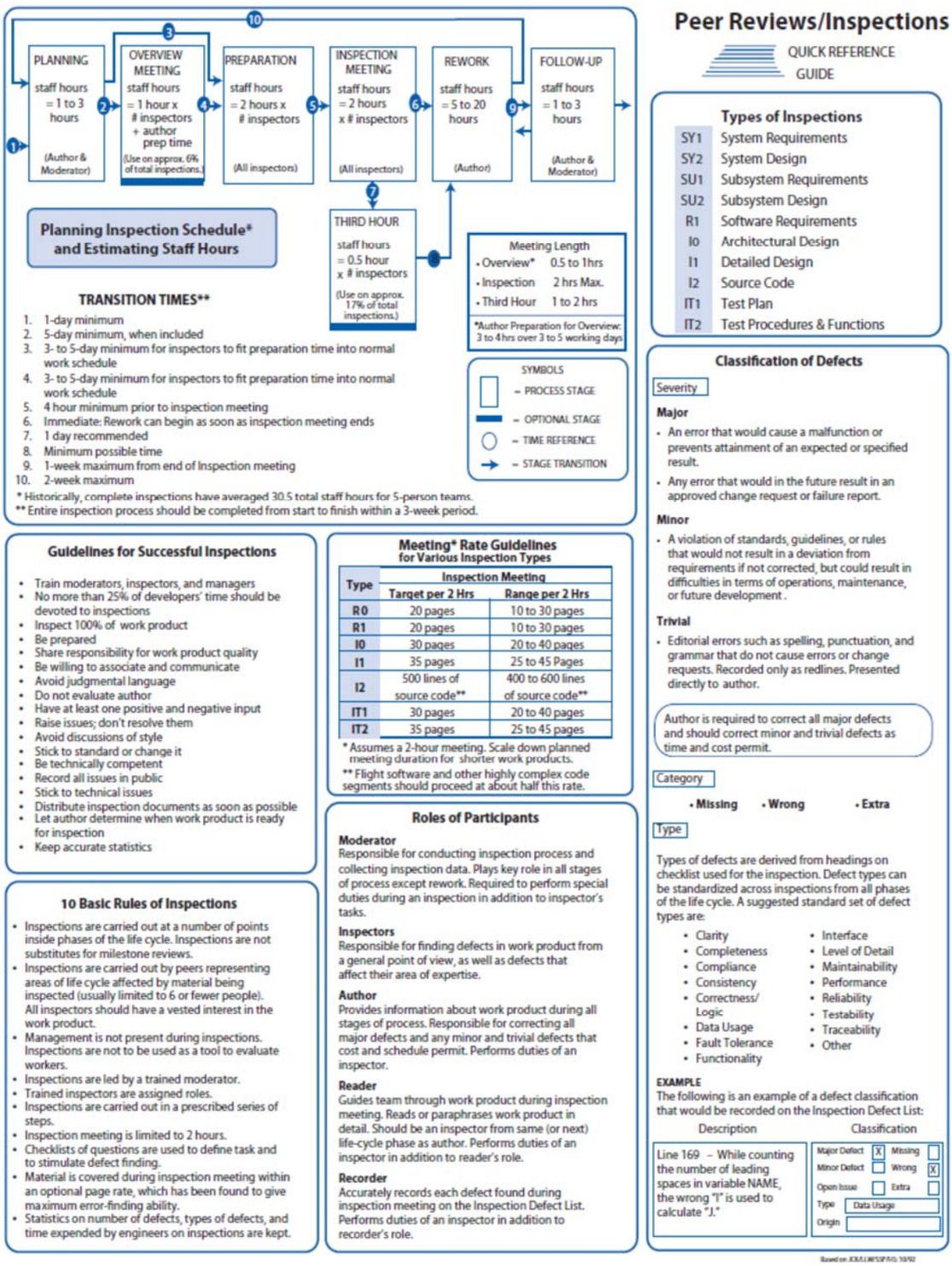


Figure N.2-2 Peer Reviews / Inspections Quick Reference Guide

Appendix O: Reserved

Appendix P: SOW Review Checklist

P.1 Editorial Checklist

- ☐ Is the SOW requirement in the form "who" shall "do what"? An example is, "The Contractor shall (perform, provide, develop, test, analyze, or other verb followed by a description of what)."

  Example SOW requirements:

  - The Contractor shall design the XYZ flight software…
  - The Contractor shall operate the ABC ground system…
  - The Contractor shall provide maintenance on the following…
  - The Contractor shall report software metrics monthly…
  - The Contractor shall integrate the PQR instrument with the spacecraft…

- ☐ Is the SOW requirement a simple sentence that contains only one requirement? Compound sentences that contain more than one SOW requirement need to be split into multiple simple sentences. (For example, "The Contractor shall do ABC and perform XYZ" should be rewritten as "The Contractor shall do ABC" and "The Contractor shall perform XYZ.")
- ☐ Is the SOW composed of simple, cohesive paragraphs, each covering a single topic? Paragraphs containing many requirements should be divided into subparagraphs for clarity.
- ☐ Has each paragraph and subparagraph been given a unique number or letter identifier? Is the numbering or lettering correct?
- ☐ Is the SOW requirement in the active rather than the passive voice? Passive voice leads to vague statements. (For example, state, "The Contractor shall hold monthly management review meetings" instead of "Management review meeting shall be held monthly.")
- ☐ Is the SOW requirement stated positively as opposed to negatively? (Replace statements such as, "The Contractor shall not exceed the budgetary limits specified" with "The contractor shall comply with the budgetary limits specified.")
- ☐ Is the SOW requirement grammatically correct?
- ☐ Is the SOW requirement free of typos, misspellings, and punctuation errors?
- ☐ Have all acronyms been defined in an acronym list or spelled out in the first occurrence?
- ☐ Have the quantities, delivery schedules, and delivery method been identified for each deliverable within the SOW or in a separate attachment/section?
- ☐ Has the content of documents to be delivered been defined in a separate attachment/section and submitted with the SOW?
- ☐ Has the file format of each electronic deliverable been defined (e.g., Microsoft—Project, Adobe—Acrobat PDF, National Instruments—Labview VIs)?

P.2 Content Checklist

- ☐ Are correct terms used to define the requirements?
  - **Shall** = requirement (binds the contractor)
  - **Should** = goal (leaves decision to contractor; avoid using this word)
  - **May** = allowable action (leaves decision to contractor; avoid using this word)
  - **Will** = facts or declaration of intent by the Government (use only in referring to the Government)
  - **Present tense** (e.g., "is") = descriptive text only (avoid using in requirements statements; use "shall" instead)
  - **NEVER** use "must"
- ☐ Is the scope of the SOW clearly defined? Is it clear what you are buying?
  Is the flow and organizational structure of the document logical and understandable? (See LPR 5000.2 "Procurement Initiator's Guide," Section 12, for helpful hints.) Is the text compatible with the title of the section it is under? Are subheadings compatible with the subject matter of headings?
- ☐ Is the SOW requirement clear and understandable?
  - ☐ Can the sentence be understood only one way?
  - ☐ Will all terminology used have the same meaning to different readers without definition? Has any terminology for which this is not the case been defined in the SOW; e.g., in a definitions section or glossary?
  - ☐ Is it free from indefinite pronouns ("this," "that," "these," "those") without clear antecedents (e.g., replace statements such as, "These shall be inspected on an annual basis" with "The fan blades shall be inspected on an annual basis")?
  - ☐ Is it stated concisely?
- ☐ Have all redundant requirements been removed? Redundant requirements can reduce clarity, increase ambiguity, and lead to contradictions.
- ☐ Is the requirement consistent with other requirements in the SOW, without contradicting itself, without using the same terminology with different meanings, without using different terminology for the same thing?
- ☐ If the SOW includes the delivery of a product (as opposed to a services-only SOW):
  - ☐ Are the technical product requirements in a separate section or attachment, apart from the activities that the contractor is required to perform? The intent is to clearly delineate between the technical product requirements and requirements for activities the contractor is to perform (e.g., separate SOW statements "The contractor shall" from technical product requirement statements such as "The system shall" and "The software shall").
  - ☐ Are references to the product and its subelements in the SOW at the level described in the technical product requirements?
  - ☐ Is the SOW consistent with and does it use the same terminology as the technical product requirements?
- ☐ Is the SOW requirement free of ambiguities? Make sure the SOW requirement is free of vague terms (for example, "as appropriate," "any," "either," "etc.," "and/or," "support," "necessary," "but not limited to," "be capable of," "be able to").
- ☐ Is the SOW requirement verifiable? Make sure the SOW requirement is free of unverifiable terms (for example, "flexible," "easy," "sufficient," "safe," "ad hoc," "adequate," "accommodate," "user-friendly," "usable," "when required," "if required," "appropriate," "fast," "portable," "lightweight," "small," "large," "maximize," "minimize," "optimize," "sufficient," "robust," "quickly," "easily," "clearly," other "ly" words, other "ize" words).

- ☐ Is the SOW requirement free of implementation constraints? SOW requirements should state WHAT the contractor is to do, NOT HOW they are to do it (for example, "The contractor shall design the XYZ flight software" states WHAT the contractor is to do, while "The contractor shall design the XYZ software using object-oriented design" states HOW the contractor is to implement the activity of designing the software. In addition, too low a level of decomposition of activities can result in specifying how the activities are to be done, rather than what activities are to be done).
- ☐ Is the SOW requirement stated in such a way that compliance with the requirement is verifiable? Do the means exist to measure or otherwise assess its accomplishment? Can a method for verifying compliance with the requirement be defined (e.g., described in a quality assurance surveillance plan)?
- ☐ Is the background material clearly labeled as such (i.e., included in the background section of the SOW if one is used)?
- ☐ Are any assumptions able to be validated and restated as requirements? If not, the assumptions should be deleted from the SOW. Assumptions should be recorded in a document separate from the SOW.
- ☐ Is the SOW complete, covering all of the work the contractor is to do?
  - ☐ Are all of the activities necessary to develop the product included (e.g., system, software, hardware, and human activities for the following: requirements, architecture, and design development; implementation and manufacturing; verification and validation; integration testing and qualification testing)?
  - ☐ Are all safety, reliability, maintainability (e.g., mean time to restore), availability, quality assurance, and security requirements defined for the total life of the contract?
  - ☐ Does the SOW include a requirement for the contractor to have a quality system (e.g., ISO certified) if one is needed?
  - ☐ Are all of the necessary management and support requirements included in the SOW (for example, project management; configuration management; systems engineering; system integration and test; risk management; interface definition and management; metrics collection, reporting, analysis, and use; acceptance testing; NASA Independent Verification and Validation (IV&V) support tasks)?
  - ☐ Are clear performance standards included and sufficient to measure contractor performance (e.g., systems, software, hardware, and service performance standards for schedule, progress, size, stability, cost, resources, and defects)? See Langley's *Guidance on System and Software Metrics for Performance-Based Contracting* for more information and examples on performance standards.
  - ☐ Are all of the necessary service activities included (for example, transition to operations, operations, maintenance, database administration, system administration, and data management)?
  - ☐ Are all of the Government surveillance activities included (for example, project management meetings; decision points; requirements and design peer reviews for systems, software, and hardware; demonstrations; test readiness reviews; other desired meetings (e.g., technical interchange meetings); collection and delivery of metrics for systems, software, hardware, and services (to provide visibility into development progress and cost); electronic access to technical and management data; and access to subcontractors and other team members for the purposes of communication)?
  - ☐ Are the Government requirements for contractor inspection and testing addressed if necessary?

- ☐ Are the requirements for contractor support of Government acceptance activities addressed if necessary?

- ☐ Does the SOW only include contractor requirements? It should not include Government requirements.
- ☐ Does the SOW give contractors full management responsibility and hold them accountable for the end result?
- ☐ Is the SOW sufficiently detailed to permit a realistic estimate of cost, labor, and other resources required to accomplish each activity?
- ☐ Are all deliverables identified (e.g., status, financial, product deliverables)? The following are examples of deliverables that are sometimes overlooked: management and development plans; technical progress reports that identify current work status, problems and proposed corrective actions, and planned work; financial reports that identify costs (planned, actual, projected) by category (e.g., software, hardware, quality assurance); products (e.g., source code, maintenance/user manual, test equipment); and discrepancy data (e.g., defect reports, anomalies).
- ☐ Does each technical and management deliverable track to a paragraph in the SOW? Each deliverable should have a corresponding SOW requirement for its preparation (i.e., the SOW identifies the title of the deliverable in parentheses after the task requiring the generation of the deliverable).
- ☐ Are all reference citations complete?
  - ☐ Are the complete number, title, and date or version of each reference specified?
  - ☐ Does the SOW reference the standards and other compliance documents in the proper SOW paragraphs?
  - ☐ Is the correct reference document cited and is it referenced at least once?
  - ☐ Is the reference document either furnished with the SOW or available at a location identified in the SOW?
  - ☐ If the referenced standard or compliance document is only partially applicable, does the SOW explicitly and unambiguously reference the portion that is required of the contractor?

Appendix Q: Reserved

Appendix R: HSI Plan Content Outline

R.1 HSI Plan Overview

The Human Systems Integration (HSI) Plan documents the strategy for and planned implementation of HSI through a particular program's/project's life cycle. The intent of HSI is:

- To ensure the human elements of the total system are effectively integrated with hardware and software elements,
- To ensure all human capital required to develop and operate the system is accounted for in life-cycle costing, and
- To ensure that the system is built to accommodate the characteristics of the user population that will operate, maintain, and support the system.

The HSI Plan is specific to a program or project and applies to NASA systems engineering per NPR 7123.1, NASA Systems Engineering Processes and Requirements. The HSI Plan should address the following:

- Roles and responsibilities for integration across HSI domains;
- Roles and responsibilities for coordinating integrated HSI domain inputs with the program team and stakeholders;
- HSI goals and deliverables for each phase of the life cycle;
- Entry and exit criteria with defined metrics for each phase, review, and milestone;
- Planned methods, tools, requirements, processes, and standards for conducting HSI;
- Strategies for identifying and resolving HSI risks; and
- Alignment strategy with the SEMP.

The party or parties responsible for program/project HSI implementation—e.g., an HSI integrator (or team)—should be identified by the program/project manager. The HSI integrator or team develops and maintains the HSI Plan with support from and coordination with the project manager and systems engineer.

Implementation of HSI on a program/project utilizes many of the tools and products already required by systems engineering; e.g., development of a ConOps, clear functional allocation across the elements of a system (hardware, software, and human), and the use of key performance measurements through the life cycle to validate and verify HSI's effectiveness. It is not the intent of the HSI Plan or its implementation to duplicate other systems engineering plans or processes, but rather to define the uniquely HSI effort being made to ensure the human element is given equal consideration to hardware/software elements of a program/project.

R.2 HSI Plan Content Outline

Each program/project-specific HSI Plan should be tailored to fit the program/project's size, scope, and purpose. The following is an example outline for a major program; e.g., space flight or aeronautics.

1.0 Introduction

1.1 Purpose

This section briefly identifies the ultimate objectives for this program/project's HSI Plan. This section also introduces the intended implementers and users of this HSI Plan.

1.2 Scope

This section describes the overall scope of the HSI Plan's role in documenting the strategy for and implementation of HSI. Overall, this section describes that the HSI Plan:

- *Is a dynamic document that will be updated at key life-cycle milestones.*
- *Is a planning and management guide that describes how HSI will be relevant to the program/project's goals.*
- *Describes planned HSI methodology, tools, schedules, and deliverables.*
- *Identifies known program/project HSI issues and concerns and how their resolutions will be addressed.*
- *Defines program/project HSI organizational elements, roles, and responsibilities.*
- *May serve as an audit trail that documents HSI data sources, analyses, activities, trade studies, and decisions not captured in other program/project documentation.*

1.3 Definitions

This section defines key HSI terms and references relevant program/project-specific terms.

2.0 Applicable Documents

This section lists all documents, references, and data sources that are invoked by HSI's implementation on the program/project, that have a direct impact on HSI outcomes, and/or are impacted by the HSI effort.

3.0 HSI Objectives

3.1 System Description

This section describes the system, missions to be performed, expected operational environment(s), predecessor and/or legacy systems (and lessons learned), capability gaps, stage of development, etc. Additionally, reference should be made to the acquisition strategy for the system; e.g., if it is developed in-house within NASA or if major systems are intended for external procurement. The overall strategy for program integration should be referenced.

Note that this information is likely captured in other program/project documentation and can be referenced in the HSI Plan rather than repeated.

3.2 HSI Relevance

At a high level, this section describes HSI's relevance to the program/project; i.e., how the HSI strategy will improve the program/project's outcome. Known HSI challenges should be described along with mention of areas where human performance in the system's operations is predicted to directly impact the probability of overall system performance and mission success.

HSI Relevance

Key Points

- *Describe performance characteristics of the human elements known to be key drivers to a desired total system performance outcome.*
- *Describe the total system performance goals that require HSI support.*
- *Identify HSI concerns with legacy systems; e.g., if operations and logistics, manpower, skill selection, required training, logistics support, operators' time, maintenance, and/or risks to safety and success exceeded expectations.*
- *Identify potential cost, schedule, risk, and trade-off concerns with the integration of human elements; e.g., quantity and skills of operators, maintainers, ground controllers, etc.*

4.0 HSI Strategy

4.1 HSI Strategy Summary

This section summarizes the HSI approaches, planning, management, and strategies for the program/project. It should describe how HSI products will be integrated across all HSI domains and how HSI inputs to program/project systems engineering and management processes contribute to system performance and help contain life-cycle cost. This section (or Implementation Summary, Section 6 of this outline) should include a top-level schedule showing key HSI milestones.

HSI Strategy

Key Points

- *Identify critical program/project-specific HSI key decision points that will be used to track HSI implementation and success.*
- *Identify key enabling (and particularly, emerging) technologies and methodologies that may be overlooked in hardware/software systems trade studies but that may positively contribute to HSI implementation; e.g., in the areas of human performance, workload, personnel management, training, safety, and survivability.*
- *Describe HSI products that will be integrated with program/project systems engineering products, analyses, risks, trade studies, and activities.*
- *Describe efforts to ensure HSI will contribute in critically important Phase A and Pre-Phase A cost-effective design concept studies.*
- *Describe the plan and schedule for updating the HSI Plan through the program / project life cycle.*

4.2 HSI Domains

This section identifies the HSI domains (see Table 2.6-1, NASA HSI Domains, in this guide) applicable to the program/project including rationale for their relevance.

HSI Domains

Key Points

- *Identify any domain(s) associated with human performance capabilities and limitations whose integration into the program/project is likely to directly affect the probability of successful program/project outcome.*
- *An overview of processes to apply, document, validate, evaluate, and mitigate HSI domain knowledge and to integrate domain knowledge into integrated HSI inputs to program/project and systems engineering processes.*

5.0 HSI Requirements, Organization, and Risk Management

5.1 HSI Requirements

This section references HSI requirements and standards applicable to the program/project and identifies the authority that invokes them; e.g., the NASA Procedural Requirements (NPR) document(s) that invoke applicability.

HSI Requirements

Key Points

- *Describe how HSI requirements that are invoked on the program/project contribute to mission success, affordability, operational effectiveness, and safety.*
- *HSI should include requirements that influence the system design to moderate manpower (operators, maintainers, system administrative, and support personnel), required skill sets (occupational specialties with high aptitude or skill requirements), and training requirements.*
- *Define the program/project-specific HSI strategy derived from NASA-STD-3001, NASA Space Flight Human-System Standard, Volume 2: Human Factors, Habitability, and Environmental Health, Standard 3.5 [V2 3005], "Human-Centered Design Process", if applicable.*
- *Capture the development process and rationale for any program/project-specific requirements not derived from existing NASA standards. In particular, manpower, skill set, and training HSI requirements/goals may be so program/project-specific as to not have NASA parent standards or requirements.*
- *Identify functional connections between HSI measures of effectiveness used to verify requirements and key performance measures used throughout the life cycle as indicators of overall HSI effectiveness.*

5.2 HSI Organization, Roles, and Responsibilities

In this section, roles and responsibilities for program/project personnel assigned to facilitate and/or manage HSI tasks are defined; e.g., the HSI integrator (and/or team if required by NPR 8705.2). HSI integrator/team functional responsibilities to the program are described in addition to identification of organizational elements with HSI responsibilities. Describe the relationships between HSI integrator/team, stakeholders, engineering technical teams, and governing bodies (control boards).

5.2.1 HSI Organization

- *Describe the HSI management structure for the program/project and identify its leaders and membership.*

- *Reference the organizational structure of the program (including industry partners) and describe the roles and responsibilities of the HSI integrator/team within that structure. Describe the HSI responsible party's relationship to other teams, including those for systems engineering, logistics, risk management, test and evaluation, and requirements verification.*
- *Provide the relationship of responsible HSI personnel to NASA Technical Authorities (Engineering, Safety, and Health/Medical).*
- *Identify if the program/project requires NASA- (Government) and/or contractor-issued HSI Plans, and identify the responsible author(s). Describe how NASA's HSI personnel will monitor and assess contractor HSI activities. For contractor-issued HSI Plans, identify requirements and processes for NASA oversight and evaluation of HSI efforts by subcontractors.*

5.2.2 HSI Roles & Responsibilities

- *Describe the HSI responsible personnel's functional responsibilities to the program / project, addressing (as examples) the following:*
  - *developing HSI program documentation;*
  - *validating human performance requirements;*
  - *conducting HSI analyses;*
  - *designing human machine interfaces to provide the level of human performance required for operations, maintenance, and support, including conduct of training;*
  - *describing the role of HSI experts in documenting and reporting the results from tests and evaluations.*
- *Define how collaboration will be performed within the HSI team, across program / project integrated product teams and with the program/project manager and systems engineer.*
- *Define how the HSI Plan and the SEMP will be kept aligned with each other.*
- *Define responsibility for maintaining and updating the HSI Plan through the program/project's life cycle.*

5.3 HSI Issue and Risk Processing

This section describes any HSI-unique processes for identifying and mitigating human system risks. HSI risks should be processed in the same manner and system as other program / project risks (technical, programmatic, schedule). However, human system risks may only be recognized by HSI domain and integration experts. Therefore, it may be important to document any unique procedures by which the program/project HSI integrator/team identifies, validates, prioritizes, and tracks the status of HSI-specific risks through the program/project risk management system. Management of HSI risks may be deemed the responsibility of the program's/project's HSI integrator/team in coordination with overall program/project risk management.

- *Ensure that potential cost, schedule, risk, and trade-off concerns with the integration of human elements (operators, maintainers, ground controllers, etc.) with the total system are identified and mitigated.*

- *Ensure that safety, health, or survivability concerns that arise as the system design and implementation emerge are identified, tracked, and managed.*
- *Identify and describe any risks created by limitations on the overall program/project HSI effort (time, funding, insufficient availability of information, availability of expertise, etc.).*
- *Describe any unique attributes of the process by which the HSI integrator/team elevates HSI risks to program/project risks.*
- *Describe any HSI-unique aspects of how human system risk mitigation strategies are deemed effective.*

6.0 HSI Implementation

6.1 HSI Implementation Summary

This section summarizes the HSI implementation approach by program/project phase. This section shows how an HSI strategy for the particular program/project is planned to be tactically enabled; i.e., establishment of HSI priorities; description of specific activities, tools, and products planned to ensure HSI objectives are met; application of technology in the achievement of HSI objectives; and an HSI risk processing strategy that identifies and mitigates technical and schedule concerns when they first arise.

HSI Implementation

Key Points

- *Relate HSI strategic objectives to the technical approaches planned for accomplishing these objectives.*
- *Overlay HSI milestones—e.g., requirements definition, verification, known trade studies, etc.—on the program/project schedule and highlight any inconsistencies, conflicts, or other expected schedule challenges.*
- *Describe how critical HSI key decision points will be dealt with as the program / project progresses through its life cycle. Indicate the plan to trace HSI key performance measures through the life cycle; i.e., from requirements to human/system functional performance allocations, through design, test, and operational readiness assessment.*
- *Identify HSI-unique systems engineering processes—e.g., verification using human-in-the-loop evaluations—that may require special coordination with program/project processes.*
- *As the system emerges, indicate plans to identify HSI lessons learned from the application of HSI on the program/project.*
- *Include a high-level summary of the resources required.*

6.2 HSI Activities and Products

In this section, map activities, resources, and products associated with planned HSI technical implementation to each systems engineering phase of the program/project. Consideration might be given to mapping the needs and products of each HSI domain by program/project phase. Examples of HSI activities include analyses, mockup/prototype human-in-the-loop evaluations, simulation/modeling, participation in design and design reviews, formative evaluations, technical interchanges, and trade studies. Examples of HSI resources include

acquisition of unique/specific HSI skill sets and domain expertise, facilities, equipment, test articles, specific time allocations, etc.

When activities, products, or risks are tied to life-cycle reviews, they should include a description of the HSI entrance and exit criteria to clearly define the boundaries of each phase, as well as resource limitations that may be associated with each activity or product (time, funding, data availability, etc.). A high-level, summary example listing of HSI activities, products, and known risk mitigations by life-cycle phase is provided in Table R.2-1.

Table R.2-1 HSI Activity, Product, or Risk Mitigation by Program/Project Phase

Life-Cycle Phase	Phase Description	Activity, Product, or Risk Mitigation
Pre-Phase A	*Concept Studies*	*ConOps (Preliminary--to include training, maintenance, logistics, etc.)*
Phase A	*Concept & Technology Development*	*HSI Plan (baseline)* *ConOps (initial)* *HSI responsible party(ies) and/or team identified before SRR* *Develop mockup(s) for HSI evaluations* *Crew Workload Evaluation Plan* *Functional allocation, crew task lists* *Validation of ConOps (planning)*
Phase B	*Preliminary Design & Technology Completion*	*HSI Plan (update)* *ConOps (baseline)* *Develop engineering-level mockup(s) for HSI evaluations* *Define crew environmental and crew health support needs (e.g., aircraft flight decks, human space flight missions)* *Assess operator interfaces through task analyses (for, e.g., aircraft cockpit operations, air traffic management, spacecraft environments, mission control for human space flight missions)* *Human-in-the-loop usability plan* *Human-rating report for PDR*
Phase C	*Final Design & Fabrication*	*HSI Plan (update)* *First Article HSI Tests* *Human-rating report for CDR*
Phase D	*System Assembly, Integ. & Test, Launch & Checkout*	*Human-rating report for ORR* *Validation of human-centered design activities* *Validation of ConOps*
Phase E	*Operations & Sustainment*	*Monitoring of human-centered design performance*
Phase F	*Closeout*	*Lessons learned report*

6.3 HSI Plan Update

The HSI Plan should be updated throughout the program/project's life-cycle management and systems engineering processes at key milestones. Milestones recommended for HSI Plan updates are listed in appendix G of NPR 7123.1, NASA Systems Engineering Processes and Requirements.

HSI Plan Updates

Key points to be addressed in each update

- *Identify the current program/project phase, the publication date of the last iteration of the HSI Plan, and the HSI Plan version number. Update the HSI Plan revision history.*
- *Describe the HSI entrance criteria for the current phase and describe any unfinished work prior to the current phase.*
- *Describe the HSI exit criteria for the current program/project phase and the work that must be accomplished to successfully complete the current program/project phase.*

Appendix S: Concept of Operations Annotated Outline

This Concept of Operations (ConOps) annotated outline describes the type and sequence of information that should be contained in a ConOps, although the exact content and sequence will be a function of the type, size, and complexity of the project. The text in italics describes the type of information that would be provided in the associated subsection. Additional subsections should be added as necessary to fully describe the envisioned system.

Cover Page

Table of Contents

1.0 Introduction

1.1 Project Description

This section will provide a brief overview of the development activity and system context as delineated in the following two subsections.

1.1.1 Background

Summarize the conditions that created the need for the new system. Provide the high-level mission goals and objective of the system operation. Provide the rationale for the development of the system.

1.1.2 Assumptions and Constraints

State the basic assumptions and constraints in the development of the concept. For example, that some technology will be matured enough by the time the system is ready to be fielded, or that the system has to be provided by a certain date in order to accomplish the mission.

1.2 Overview of the Envisioned System

This section provides an executive summary overview of the envisioned system. A more detailed description will be provided in Section 3.0

1.2.1 Overview

This subsection provides a high-level overview of the system and its operation. Pictorials, graphics, videos, models, or other means may be used to provide this basic understanding of the concept.

1.2.2 System Scope

This section gives an estimate of the size and complexity of the system. It defines the system's external interfaces and enabling systems. It describes what the project will encompass and what will lie outside of the project's development.

2.0 Documents

2.1 Applicable Documents

This section lists all the documents, models, standards or other material that are applicable and some or all of which will form part of the requirements of the project.

2.2 Reference Documents

This section provides supplemental information that might be useful in understanding the system or its scenarios.

3.0 Description of Envisioned System

This section provides a more detailed description of the envisioned system and its operation as contained in the following subsections.

3.1 Needs, Goals and Objectives of Envisioned System

This section describes the needs, goals, and objectives as expectations for the system capabilities, behavior, and operations. It may also point to a separate document or model that contains the current up-to-date agreed-to expectations.

3.2 Overview of System and Key Elements

This section describes at a functional level the various elements that will make up the system, including the users and operators. These descriptions should be implementation free; that is, not specific to any implementation or design but rather a general description of what the system and its elements will be expected to do. Graphics, pictorials, videos, and models may be used to aid this description.

3.3 Interfaces

This section describes the interfaces of the system with any other systems that are external to the project. It may also include high-level interfaces between the major envisioned elements of the system. Interfaces may include mechanical, electrical, human user/operator, fluid, radio frequency, data, or other types of interactions.

3.4 Modes of Operations

This section describes the various modes or configurations that the system may need in order to accomplish its intended purpose throughout its life cycle. This may include modes needed in the development of the system, such as for testing or training, as well as various modes that will be needed during it operational and disposal phases.

3.5 Proposed Capabilities

This section describes the various capabilities that the envisioned system will provide. These capabilities cover the entire life cycle of the system's operation, including special capabilities needed for the verification/validation of the system, its capabilities during its intended operations, and any special capabilities needed during the decommissioning or disposal process.

4.0 Physical Environment

This section should describe the environment that the system will be expected to perform in throughout its life cycle, including integration, tests, and transportation. This may include expected and off-nominal temperatures, pressures, radiation, winds, and other atmospheric, space, or aquatic conditions. A description of whether the system needs to operate, tolerate with degraded performance, or just survive in these conditions should be noted.

5.0 Support Environment

This section describes how the envisioned system will be supported after being fielded. This includes how operational planning will be performed and how commanding or other uploads

will be determined and provided, as required. Discussions may include **how** *the envisioned system would be maintained, repaired, replaced, it's sparing philosophy, and how future upgrades may be performed. It may also include assumptions on the level of continued support from the design teams.*

6.0 Operational Scenarios, Use Cases and/or Design Reference Missions

This section takes key scenarios, use cases, or DRM and discusses what the envisioned system provides or how it functions throughout that single-thread timeline. The number of scenarios, use cases, or DRMs discussed should cover both nominal and off-nominal conditions and cover all expected functions and capabilities. A good practice is to label each of these scenarios to facilitate requirements traceability; e.g., [DRM-0100], [DRM-0200], etc.

6.1 Nominal Conditions

These scenarios, use cases, or DRMs cover how the envisioned system will operate under normal circumstances where there are no problems or anomalies taking place.

6.2 Off-Nominal Conditions

These scenarios cover cases where some condition has occurred that will need the system to perform in a way that is different from normal. This would cover failures, low performance, unexpected environmental conditions, or operator errors. These scenarios should reveal any additional capabilities or safeguards that are needed in the system.

7.0 Impact Considerations

This section describes the potential impacts, both positive and negative, on the environment and other areas.

7.1 Environmental Impacts

Describes how the envisioned system could impact the environment of the local area, state, country, worldwide, space, and other planetary bodies as appropriate for the systems intended purpose. This includes the possibility of the generation of any orbital debris, potential contamination of other planetary bodies or atmosphere, and generation of hazardous wastes that will need disposal on earth and other factors. Impacts should cover the entire life cycle of the system from development through disposal.

7.2 Organizational Impacts

Describes how the envisioned system could impact existing or future organizational aspects. This would include the need for hiring specialists or operators, specialized or widespread training or retraining, and use of multiple organizations.

7.3 Scientific/Technical Impacts

This subsection describes the anticipated scientific or technical impact of a successful mission or deployment, what scientific questions will be answered, what knowledge gaps will be filled, and what services will be provided. If the purpose of this system is to improve operations or logistics instead of science, describe the anticipated impact of the system in those terms.

8.0 Risks and Potential Issues

This section describes any risks and potential issues associated with the development, operations or disposal of the envisioned system. Also includes concerns/risks with the project schedule,

staffing support, or implementation approach. Allocate subsections as needed for each risk or issue consideration. Pay special attention to closeout issues at the end of the project.

Appendix A Acronyms

This part lists each acronym used in the ConOps and spells it out.

Appendix B Glossary of Terms

The part lists key terms used in the ConOps and provides a description of their meaning.

Appendix T: Systems Engineering in Phase E

T.1 Overview

In general, normal Phase E activities reflect a reduced emphasis on system design processes but a continued focus on product realization and technical management. Product realization process execution in Phase E takes the form of continued mission plan generation (and update), response to changing flight conditions (and occurrence of in-flight anomalies), and update of mission operations techniques, procedures, and guidelines based on operational experience gained. Technical management processes ensure that appropriate rigor and risk management practices are applied in the execution of the product realization processes.

Successful Phase E execution requires the prior establishment of mission operations capabilities in four (4) distinct categories: tools, processes, products, and trained personnel. These capabilities may be developed as separate entities, but need to be fused together in Phase E to form an end-to-end operational capability.

Although systems engineering activities and processes are constrained throughout the entire project life cycle, additional pressures exist in Phase E:

- Increased resource constraints – Even when additional funding or staffing can be secured, building new capabilities or training new personnel may require more time or effort than is available. Project budget and staffing profiles generally decrease at or before entry into Phase E, and the remaining personnel are typically focused on mission execution.
- Unforgiving schedule – Unlike pre-flight test activities, it may be difficult or even impossible to pause mission execution to deal with technical issues of a spacecraft in operation. It is typically difficult or impossible to truly pause mission execution after launch.

These factors must be addressed when considering activities that introduce change and risk during Phase E.

Note: When significant hardware or software changes are required in Phase E, the logical decomposition process may more closely resemble that exercised in earlier project phases. In such cases, it may be more appropriate to identify the modification as a new project executing in parallel – and coordinated with – the operating project.

T.2 Transition from Development to Operations

An effective transition from development to operations phases requires prior planning and coordination among stakeholders. This planning should focus not only on the effective transition of hardware and software systems into service but also on the effective transfer of knowledge, skills, experience, and processes into roles that support the needs of flight operations.

Development phase activities need to clearly and concisely document system knowledge in the form of operational techniques, characteristics, limits, and constraints – these are key inputs used by flight operations personnel in building operations tools and techniques. Phase D Integration and Test (I&T) activities share many common needs with Phase E operations activities. Without prior planning and agreement, however, similar products used in these two phases may be formatted so differently that one set cannot be used for both purposes. The associated product

duplication is often unexpected and results in increased cost and schedule risk. Instead, system engineers should identify opportunities for product reuse early in the development process and establish common standards, formats, and content expectations to enable transition and reuse.

Similarly, the transfer of skills and experience should be managed through careful planning and placement of key personnel. In some cases, key design, integration, and test personnel may be transitioned into the mission operations team roles. In other cases, dedicated mission operations personnel may be assigned to shadow or assist other teams during Phase A-D activities. In both cases, assignees bring knowledge, skills, and experience into the flight operations environment. Management of this transition process can, however, be complex as these personnel may be considered key to both ongoing I&T and preparation for upcoming operations. Careful and early planning of personnel assignments and transitions is key to success in transferring skills and experience.

T.3 System Engineering Processes in Phase E

T.3.1 System Design Processes

In general, system design processes are complete well before the start of Phase E. However, events during operations may require that these processes be revisited in Phase E.

T.3.1.1 Stakeholder Expectations Definition

Stakeholder expectations should have been identified during development phase activities, including the definition of operations concepts and design reference missions. Central to this definition is a consensus on mission success criteria and the priority of all intended operations. The mission operations plan should state and address these stakeholder expectations with regard to risk management practices, planning flexibility and frequency of opportunities to update the plan, time to respond and time/scope of status communication, and other key parameters of mission execution. Additional detail in the form of operational guidelines and constraints should be incorporated in mission operations procedures and flight rules.

The Operations Readiness Review (ORR) should confirm that stakeholders accept the mission operations plan and operations implementation products.

However, it is possible for events in Phase E to require a reassessment of stakeholder expectations. Significant in-flight anomalies or scientific discoveries during flight operations may change the nature and goals of a mission. Mission systems engineers, mission operations managers, and program management need to remain engaged with stakeholders throughout Phase E to identify potential changes in expectations and to manage the acceptance or rejection of such changes during operations.

T.3.1.2 Technical Requirements Definition

New technical requirements and changes to existing requirements may be identified during operations as a result of:

- New understanding of system characteristics through flight experience;
- The occurrence of in-flight anomalies; or
- Changing mission goals or parameters (such as mission extension).

These changes or additions are generally handled as change requests to an operations baseline already under configuration management and possibly in use as part of ongoing flight operations. Such changes are more commonly directed to the ground segment or operations products (operational constraints, procedures, etc.). Flight software changes may also be considered, but flight hardware changes for anything other than human-tended spacecraft are rarely possible.

Technical requirement change review can be more challenging in Phase E as fewer resources are available to perform comprehensive review. Early and close involvement of Safety and Mission Assurance (SMA) representatives can be key in ensuring that proposed changes are appropriate and within the project's allowable risk tolerance.

T.3.1.3 Logical Decomposition

In general, logical decomposition of mission operations functions is performed during development phases. Additional logical decomposition during operations is more often applied to the operations products: procedures, user interfaces, and operational constraints. The authors and users of these products are often the most qualified people to judge the appropriate decomposition of new or changed functionality as a series of procedures or similar products.

T.3.1.4 Design Solution Definition

Similar to logical decomposition, design solution definition tasks may be better addressed by those who develop and use the products. Minor modifications may be handled entirely within an operations team (with internal reviews), while larger changes or additions may warrant the involvement of program-level system engineers and Safety and Mission Assurance (SMA) personnel.

Scarcity of time and resources during Phase E can make implementation of these design solutions challenging. The design solution needs to take into account the availability of and constraints to resources.

T.3.1.5 Product Implementation

Personnel who implement mission operations products such as procedures and spacecraft command scripts should be trained and certified to the appropriate level of skill as defined by the project. Processes governing the update and creation of operations products should be in place and exercised prior to Phase E.

T.3.2 Product Realization Processes

Product realization processes in Phase E are typically executed by Configuration Management (CM) and test personnel. It is common for these people to be "shared resources;" i.e., personnel who fulfil other roles in addition to CM and test roles.

T.3.2.1 Product Integration

Product integration in Phase E generally involves bringing together multiple operations products – some preexisting and others new or modified – into a proposed update to the baseline mission operations capability.

The degree to which a set of products is integrated may vary based on the size and complexity of the project. Small projects may define a baseline – and update to that baseline – that spans the entire set of all operations products. Larger or more complex projects may choose to create logical baseline subsets divided along practical boundaries. In a geographically disperse set of separate mission operations Centers, for example, each Center may be initially integrated as a separate product. Similarly, the different functions within a single large control Center – planning, flight dynamics, command and control, etc. – may be established as separately baselined products. Ultimately, however, some method needs to be established to ensure that the product realization processes identify and assess all potential impacts of system changes.

T.3.2.2 Product Verification

Product verification in Phase E generally takes the form of unit tests of tools, data sets, procedures, and other items under simulated conditions. Such "thread tests" may exercise single specific tasks or functions. The fidelity of simulation required for verification varies with the nature and criticality of the product. Key characteristics to consider include:

- Runtime – Verification of products during flight operations may be significantly time constrained. Greater simulation fidelity can result in slower simulation performance. This slower performance may be acceptable for some verification activities but may be too constraining for others.
- Level of detail – Testing of simple plans and procedures may not require high-fidelity simulation of a system's dynamics. For example, simple state change processes may be tested on relatively low-fidelity simulations. However, operational activities that involve dynamic system attributes – such as changes in pressure, temperature, or other physical properties may require testing with much higher-fidelity simulations.
- Level of integration – Some operations may impact only a single subsystem, while others can affect multiple systems or even the entire spacecraft.
- Environmental effects – Some operations products and procedures may be highly sensitive to environmental conditions, while others may not. For example, event sequences for atmospheric entry and deceleration may require accurate weather data. In contrast, simple system reconfiguration procedures may not be impacted by environmental conditions at all.

T.3.2.3 Product Validation

Product validation is generally executed through the use of products in integrated operational scenarios such as mission simulations, operational readiness tests, and/or spacecraft end-to-end tests. In these environments, a collection of products is used by a team of operators to simulate an operational activity or set of activities such as launch, activation, rendezvous, science operations, or Entry, Descent, and Landing (EDL). The integration of multiple team members and operations products provides the context necessary to determine if the product is appropriate and meets the true operations need.

T.3.2.4 Product Transition

Transition of new operational capabilities in Phase E is generally overseen by the mission operations manager or a Configuration Control Board (CCB) chaired by the mission operations manager or the project manager.

Proper transition management includes the inspection of product test (verification and validation) results as well as the readiness of the currently operating operations system to accept changes. Transition during Phase E can be particularly challenging as the personnel using these capabilities also need to change techniques, daily practices, or other behaviors as a result. Careful attention should be paid to planned operations, such as spacecraft maneuvers or other mission critical events and risks associated with performing product transition at times near such events.

T.3.3 Technical Management Processes

Technical management processes are generally a shared responsibility of the project manager and the mission operations manager. Clear agreement between these two parties is essential in ensuring that Phase E efforts are managed effectively.

T.3.3.1 Technical Planning

Technical planning in Phase E generally focuses on the management of scarce product development resources during mission execution. Key decision-makers, including the mission operations manager and lower operations team leads, need to review the benefits of a change against the resource cost to implement changes. Many resources are shared in Phase E – for example, product developers may also serve other real-time operations roles– and the additional workload placed on these resources should be viewed as a risk to be mitigated during operations.

T.3.3.2 Requirements Management

Requirements management during Phase E is similar in nature to pre-Phase E efforts. Although some streamlining may be implemented to reduce process overhead in Phase E, the core need to review and validate requirements remains. As most Phase E changes are derived from a clearly demonstrated need, program management may reduce or waive the need for complete requirements traceability analysis and documentation.

T.3.3.3 Interface Management

It is relatively uncommon for interfaces to change in Phase E, but this can occur when a software tool is modified or a new need is uncovered. Interface definitions should be managed in a manner similar to that used in other project phases.

T.3.3.4 Technical Risk Management

Managing technical risks during operations can be more challenging during Phase E than during other phases. New risks discovered during operations may be the result of system failures or changes in the surrounding environment. Where additional time may be available to assess and mitigate risk in other project phases, the nature of flight operations may limit the time over which risk management can be executed. For this reason, every project should develop a formal process for handling anomalies and managing risk during operations. This process should be exercised before flight, and decision-makers should be well versed in the process details.

T.3.3.5 Configuration Management

Effective and efficient Configuration Management (CM) is essential during operations. Critical operations materials, including procedures, plans, flight datasets, and technical reference material

need to be secure, up to date, and easily accessed by those who make and enact mission critical decisions. CM systems – in their intended flight configuration – should be exercised as part of operational readiness tests to ensure that the systems, processes, and participants are flight-ready.

Access to such operations products is generally time-critical, and CM systems supporting that access should be managed accordingly. Scheduled maintenance or other "downtime" periods should be coordinated with flight operations plans to minimize the risk of data being inaccessible during critical activities.

T.3.3.6 Technical Data Management

Tools, procedures, and other infrastructure for Technical Data Management must be baselined, implemented, and verified prior to flight operations. Changes to these capabilities are rarely made during Phase E due to the high risk of data loss or reduction in operations efficiency when changing during operations.

Mandatory Technical Data Management infrastructure changes, when they occur, should be carefully reviewed by those who interact with the data on a regular basis. This includes not only operations personnel, but also engineering and science customers of that data.

T.3.3.7 Technical Assessment

Formal technical assessments during Phase E are typically focused on the upcoming execution of a specific operational activity such as launch, orbit entry, or decommissioning. Reviews executed while flight operations are in progress should be scoped to answer critical questions while not overburdening the project or operations team.

Technical Performance Measures (TPMs) in Phase E may differ significantly from those in other project phases. Phase E TPMs may focus on the accomplishment of mission events, the performance of the system in operation, and the ability of the operations team to support upcoming events.

T.3.3.8 Decision Analysis

The Phase E Decision Analysis Process is similar to that in other project phases but may emphasize different criteria. For example, the ability to change a schedule may be limited by the absolute timing of events such as an orbit entry or landing on a planetary surface. Cost trades may be more constrained by the inability to add trained personnel to support an activity. Technical trades may be limited by the inability to modify hardware in operation.

References Cited

This appendix contains references that were cited in the sections of this guide.

Preface

NPR 7123.1, Systems Engineering Processes and Requirements

NASA Chief Engineer and the NASA Integrated Action Team (NIAT) report, *Enhancing Mission Success -- A Framework for the Future*, December 21, 2000. Authors: McBrayer, Robert O and Thomas, Dale, NASA Marshall Space Flight Center, Huntsville, AL United States.

NASA. *Columbia Accident Investigation Board (CAIB) Report*, 6 volumes: Aug. 26, Oct. 2003. http://www.nasa.gov/columbia/caib/html/report.html

NASA. Diaz Report, *A Renewed Commitment to Excellence: An Assessment of the NASA Agency-wide Applicability of the Columbia Accident Investigation Board Report*, January 30, 2004. Mr. Al Diaz, Director, Goddard Space Flight Center, and team.

International Organization for Standardization (ISO) 9000:2015, *Quality management systems - Fundamentals and vocabulary*. Geneva: International Organization for Standardization, 2015.

Section 1.1 Purpose

NPR 7123.1. Systems Engineering Processes and Requirements

Section 1.2 Scope and Depth

NASA Office of Chief Information Officer (OCIO), *Information Technology Systems Engineering Handbook Version 2.0*

NASA-HDBK-2203, *NASA Software Engineering Handbook* (February 28, 2013)

Section 2.0 Fundamentals of Systems Engineering

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7120.7, NASA Information Technology and Institutional Infrastructure Program and Project Management Requirements

NPR 7120.8, NASA Research and Technology Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

NASA Engineering Network (NEN) Systems Engineering Community of Practice (SECoP), located at https://nen.nasa.gov/web/se

Griffin, Michael D., NASA Administrator. “System Engineering and the Two Cultures of Engineering.” Boeing Lecture, Purdue University, March 28, 2007.

Rechtin, Eberhardt. *Systems Architecting of Organizations: Why Eagles Can't Swim*. Boca Raton: CRC Press, 2000.

Section 2.1 The Common Technical Processes and the SE Engine

NPR 7123.1, NASA Systems Engineering Processes and Requirements

Society of Automotive Engineers (SAE) and the European Association of Aerospace Industries (EAAI). *AS9100C Quality Management Systems (QMS) - Requirements for Aviation, Space, and Defense Organizations* Revision C: January 15, 2009.

Section 2.3 Example of Using the SE Engine

NPD 1001.0, 2006 NASA Strategic Plan

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

Section 2.5 Cost Effectiveness Considerations

Department of Defense (DOD) Defense Acquisition University (DAU). *Systems Engineering Fundamentals Guide*. Fort Belvoir, VA, 2001.

INCOSE-TP-2003-002-04, *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*, Version 4, edited by Walden, David D., et al., 2015

Section 2.6 Human Systems Integration (HSI) in the SE Process

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

Section 3.0 NASA Program/Project Life Cycle

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7120.8, NASA Research and Technology Program and Project Management Requirements

NASA Office of the Chief Information Officer (OCIO), *Information Technology Systems Engineering Handbook* Version 2.0

NASA/SP-2014-3705, *NASA Space Flight Program and Project Management Handbook*

Section 3.1 Program Formulation

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7120.7, NASA Information Technology and Institutional Infrastructure Program and Project Management Requirements

NPR 7120.8, NASA Research and Technology Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

Section 3.2 Program Implementation

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

Section 3.3 Project Pre-Phase A: Concept Studies

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

Section 3.4 Project Phase A: Concept and Technology Development

NPD 1001.0, 2014 NASA Strategic Plan

NPR 2810.1, Security of Information Technology

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

NPR 7150.2, NASA Software Engineering Requirements

NASA-STD-8719.14, *Handbook for Limiting Orbital Debris*. Rev A with Change 1. December 8, 2011.

National Institute of Standards and Technology (NIST), Federal Information Processing Standard Publication (FIPS PUB) 199, *Standards for Security Categorization of Federal Information and Information Systems*, February 2004.

Section 3.5 Project Phase B: Preliminary Design and Technology Completion

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

Section 3.6 Project Phase C: Final Design and Fabrication

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

Section 3.7 Project Phase D: System Assembly, Integration and Test, Launch

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

NASA Office of the Chief Information Officer (OCIO), *Information Technology Systems Engineering Handbook* Version 2.0

Section 3.8 Project Phase E: Operations and Sustainment

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

Section 3.9 Project Phase F: Closeout

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

NPD 8010.3, Notification of Intent to Decommission or Terminate Operating Space Systems and Terminate Missions

NPR 8715.6, NASA Procedural Requirements for Limiting Orbital Debris

Section 3.10 Funding: The Budget Cycle

NASA's *Financial Management Requirements (FMR)* Volume 4

Section 3.11 Tailoring and Customization of NPR 7123.1 Requirements

NPD 1001.0, 2014 NASA Strategic Plan

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7120.7, NASA Information Technology and Institutional Infrastructure Program and Project Management Requirements

NPR 7120.8, NASA Research and Technology Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

NPR 7150.2, NASA Software Engineering Requirements

NPR 8705.4, Risk Classification for NASA Payloads

NASA-HDBK-2203, *NASA Software Engineering Handbook* (February 28, 2013)

NASA Engineering Network (NEN) Systems Engineering Community of Practice (SECoP), located at https://nen.nasa.gov/web/se

Section 4.1 Stakeholder Expectations Definition

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NASA Science Mission Directorate strategic plans

Presidential Policy Directive PPD-4 (2010), National Space Policy

Presidential Policy Directive PPD-21 (2013), Critical Infrastructure Security and Resilience

Ball, Robert E. (Naval Postgraduate School), *The Fundamentals of Aircraft Combat Survivability Analysis and Design*, 2nd Edition, AIAA Education Series, 2003

Larson (Wiley J.), Kirkpatrick, Sellers, Thomas, and Verma. *Applied Space Systems Engineering: A Practical Approach to Achieving Technical Baselines.* 2nd Edition, Boston, MA: McGraw-Hill Learning Solutions, CEI Publications, 2009.

Section 4.2 Technical Requirements Definition

NPR 7120.10, Technical Standards for NASA Programs and Projects

NPR 8705.2, Human-Rating Requirements for Space Systems

NPR 8715.3, NASA General Safety Program Requirements

NASA-STD-3001, *NASA Space Flight Human System Standard* - 2 volumes

NASA-STD-8719.13, *Software Safety Standard*, Rev C. Washington, DC, May 7, 2013.

NASA/SP-2010-3407, *Human Integration Design Handbook (HIDH)*

Section 4.3 Logical Decomposition

Department of Defense (DOD) Architecture Framework (DODAF) Version 2.02 Change 1, January 2015

Institute of Electrical and Electronics Engineers (IEEE) STD 610.12-1990, *IEEE Standard Glossary of Software Engineering Terminology*. Reaffirmed 2002. Superseded by ISO/IEC/IEEE 24765:2010, *Systems and Software Engineering – Vocabulary*

Section 4.4 Design Solution Definition

NPD 8730.5, NASA Quality Assurance Program Policy

NPR 8735.2, Management of Government Quality Assurance Functions for NASA Contracts

NASA-HDBK-1002, *Fault Management (FM) Handbook*, Draft 2, April 2012.

NASA-STD-3001, *NASA Space Flight Human System Standard* – 2 volumes

NASA-STD-8729.1, *Planning, Developing, and Maintaining an Effective Reliability and Maintainability (R&M) Program*. Washington, DC, December 1, 1998.

Code of Federal Regulations (CFR), Title 48 – Federal Acquisition Regulation (FAR) System, Part 46.4 Government Contract Quality Assurance (48 CFR 46.4)

International Organization for Standardization, ISO 9001:2015 *Quality Management Systems (QMS)*

Society of Automotive Engineers and the European Association of Aerospace Industries. AS9100C *Quality Management Systems (QMS) - Requirements for Aviation, Space, and Defense Organizations* Revision C: 2009-01-15

Blanchard, Benjamin S., *System Engineering Management*. 4th Edition, Hoboken, NJ: John Wiley & Sons, Inc., 2008

Section 5.1 Product Implementation

NPR 7150.2, NASA Software Engineering Requirements

NASA Engineering Network (NEN) Systems Engineering Community of Practice (SECoP), located at https://nen.nasa.gov/web/se

NASA Engineering Network (NEN) V&V Community of Practice, located at https://nen.nasa.gov/web/se

American Institute of Aeronautics and Astronautics (AIAA) G-118-2006e. *AIAA Guide for Managing the Use of Commercial Off the Shelf (COTS) Software Components for Mission-Critical Systems*. Reston, VA, 2006

Section 5.2 Product Integration

NASA Lyndon B. Johnson Space Center (JSC-60576), National Space Transportation System (NSTS), Space Shuttle Program, Transition Management Plan, May 9, 2007

Section 5.3 Product Verification

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7120.8, NASA Research and Technology Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

NPR 8705.4, Risk Classification for NASA Payloads

NASA-STD-7009, *Standard for Models and Simulations*. Washington, DC, October 18, 2013

NASA GSFC-STD-7000, Goddard Technical Standard: *General Environmental Verification Standard (GEVS) for GSFC Flight Programs and Projects*. Goddard Space Flight Center. April 2005

Department of Defense (DOD). MIL-STD-1540D, *Product Verification Requirements for Launch, Upper Stage, and Space Vehicles*. January 15, 1999

Section 5.4 Product Validation

NPD 7120.4, NASA Engineering and Program/Project Management Policy

NPR 7150.2, NASA Software Engineering Requirements

Section 5.5 Product Transition

(The) National Environmental Policy Act of 1969 (NEPA). See 42 U.S.C. 4321-4347. https://ceq.doe.gov/welcome.html

Section 6.1 Technical Planning

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPD 7120.6, Knowledge Policy on Programs and Projects

NPR 7123.1, NASA Systems Engineering Processes and Requirements

NASA-SP-2010-3403, *NASA Schedule Management Handbook*

NASA-SP-2010-3404, *NASA Work Breakdown Structure Handbook*

NASA Cost Estimating Handbook (CEH), Version 4, February 2015.

DOD. MIL-STD-881C, *Work Breakdown Structure (WBS) for Defense Materiel Items*. Washington, DC, October 3, 2011.

Institute of Electrical and Electronics Engineers (IEEE) STD 1220-2005. *IEEE Standard for Application and Management of the Systems Engineering Process*, Washington, DC, 2005.

Office of Management and Budget (OMB) Circular A-94,"Guidelines and Discount Rates for Benefit-Cost Analysis of Federal Programs" (10/29/1992)

Joint (cost and schedule) Confidence Level (JCL). Frequently asked questions (FAQs) can be found at: http://www.nasa.gov/pdf/394931main_JCL_FAQ_10_12_09.pdf

The U. S. Chemical Safety Board (CSB) case study reports on mishaps found at: http://www.csb.gov/

Section 6.3 Interface Management

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

Section 6.4 Technical Risk Management

NPR 8000.4, Agency Risk Management Procedural Requirements

NASA/SP-2010-576, *NASA Risk-Informed Decision Making Handbook*

NASA/SP-2011-3421, *Probabilistic Risk Assessment Procedures Guide for NASA Managers and Practitioners*

NASA/SP-2011-3422, *NASA Risk Management Handbook*

Code of Federal Regulations (CFR) Title 22 – Foreign Relations, Parts 120-130 Department of State: International Traffic in Arms Regulations (ITAR) (22 CFR 120-130). Implements 22

U.S.C. 2778 of the Arms Export Control Act (AECA) of 1976 and Executive Order 13637, “Administration of Reformed Export Controls," March 8, 2013

Section 6.5 Configuration Management

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NASA. *Columbia Accident Investigation Board (CAIB) Report*, 6 volumes: Aug. 26, Oct. 2003. http://www.nasa.gov/columbia/caib/html/report.html

NASA. *NOAA N-Prime Mishap Investigation Final Report*, Sept. 13, 2004 http://www.nasa.gov/pdf/65776main_noaa_np_mishap.pdf

SAE International (SAE) / Electronic Industries Alliance (EIA) 649B-2011, *Configuration Management Standard (Aerospace Sector)* April 1, 2011

American National Standards Institute (ANSI) / Electronic Industries Alliance (EIA). ANSI/EIA-649, *National Consensus Standard for Configuration Management*, 1998-1999

Section 6.6 Technical Data Management

NPR 1441.1, NASA Records Management Program Requirements

NPR 1600.1, NASA Security Program Procedural Requirements

NID 1600.55, Sensitive But Unclassified (SBU) Controlled Information

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

NASA Form (NF) 1686, NASA Scientific and Technical Document Availability Authorization (DAA) for Administratively Controlled Information.

Code of Federal Regulations (CFR) Title 22 – Foreign Relations, Parts 120-130 Department of State: International Traffic in Arms Regulations (ITAR) (22 CFR 120-130). Implements 22 U.S.C. 2778 of the Arms Export Control Act (AECA) of 1976 and Executive Order 13637, “Administration of Reformed Export Controls," March 8, 2013

The Invention Secrecy Act of 1951, 35 U.S.C. §181-§188. Secrecy of Certain Inventions and Filing Applications in Foreign Country; §181 - Secrecy of Certain Inventions and Withholding of Patent.

Code of Federal Regulations (CFR) Title 37 – Patents, Trademarks, and Copyrights; Part 5 Secrecy of Certain Inventions and Licenses to Export and File Applications in Foreign Countries; Part 5.2 Secrecy Order. (37 CFR 5.2)

Section 6.7 Technical Assessment

NPR 1080.1, Requirements for the Conduct of NASA Research and Technology (R&T)

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7120.7, NASA Information Technology and Institutional Infrastructure Program and Project Management Requirements

NPR 7120.8, NASA Research and Technology Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

NPR 8705.4, Risk Classification for NASA Payloads

NPR 8705.6, Safety and Mission Assurance (SMA) Audits, Reviews, and Assessments

NPR 8715.3, NASA General Safety Program Requirements

NASA-HDBK-2203, *NASA Software Engineering Handbook*. February 28, 2013

NASA/SP-2012-599, *NASA's Earned Value Management (EVM) Implementation Handbook*

NASA Federal Acquisition Regulation (FAR) Supplement (NFS) 1834.201, Earned Value Management System Policy.

NASA EVM website http://evm.nasa.gov/index.html

NASA Engineering Network (NEN) EVM Community of Practice located at https://nen.nasa.gov/web/pm/evm

NASA Engineering Network (NEN) Systems Engineering Community of Practice (SECoP) under Tools and Methods at https://nen.nasa.gov/web/se/tools/ and then NASA Tools & Methods

American National Standards Institute/Electronic Industries Alliance (ANSI-EIA), Standard 748-C *Earned Value Management Systems*. March, 2013.

International Council on Systems Engineering (INCOSE). INCOSE-TP-2003-020-01, *Technical Measurement*, Version 1.0, 27 December 2005. Prepared by Garry J. Roedler (Lockheed Martin) and Cheryl Jones (U.S. Army).

Section 6.8 Decision Analysis

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

Brughelli, Kevin (Lockheed Martin), Deborah Carstens (Florida Institute of Technology), and Tim Barth (Kennedy Space Center), "Simulation Model Analysis Techniques," Lockheed Martin presentation to KSC, November 2003 (see Figure 6.8-4.)

Saaty,Thomas L. *The Analytic Hierarchy Process*. New York: McGraw-Hill, 1980

Section 7.1 Engineering with Contracts

NPR 7123.1, NASA Systems Engineering Processes and Requirements

NPR 8735.2, Management of Government Quality Assurance Functions for NASA Contracts

NASA-STD-7009, *Standard for Models and Simulations*. Washington, DC, October 18, 2013.

NASA Procurement Library found at http://www.hq.nasa.gov/office/procurement/

NASA Langley Research Center (LARC) *Guidance on System and Software Metrics for Performance-Based Contracting*. 2013 sites-e.larc.nasa.gov/sweng/files/2013/05/Guidance_on_Metrics_for_PBC_R1V01.doc

Section 7.2 Concurrent Engineering Methods

Deming, W. Edwards, see https://www.deming.org/

Karpati, G., Martin, J., Steiner, M., Reinhardt, K., "The Integrated Mission Design Center (IMDC) at NASA Goddard Space Flight Center," *IEEE Aerospace Conference 2003 Proceedings*, Volume 8, Page(s): 8_3657 - 8_3667, 2003

Kluger, Jeffrey with Dan Cray, "Management Tips from the Real Rocket Scientists," *Time Magazine*, November 2005

McGuire, M., Oleson, S., Babula, M., and Sarver-Verhey, T., Concurrent Mission and Systems Design at NASA Glenn Research Center: The origins of the COMPASS Team, *AIAA Space 2011 Proceedings*, September 27-29, 2011, Long Beach, CA

Moeller, Robert C., Chester Borden, Thomas Spilker, William Smythe, Robert Lock , "Space Missions Trade Space Generation and Assessment using the JPL Rapid Mission Architecture (RMA) Team Approach", *IEEE Aerospace Conference*, Big Sky, Montana, Mar 2011

Mulqueen, J.; R. Hopkins; D. Jones, "The MSFC Collaborative Engineering Process for Preliminary Design and Concept Definition Studies." 2012 http://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20120001572.pdf

Oberto, R.E., Nilsen, E., Cohen, R., Wheeler, R., DeFlorio, P., and Borden, C., "The NASA Exploration Design Team; Blueprint for a New Design Paradigm", *2005 IEEE Aerospace Conference*, Big Sky, Montana, March 2005

Pennell, J. and Winner, R., "Concurrent Engineering: Practices and Prospects", *Global Telecommunications Conference, (GLOBECOM '89)*, 1989

Wall, S., "Use of Concurrent Engineering in Space Mission Design," *Proceedings of EuSEC 2000*, Munich, Germany, September 2000

Warfield, K., "Addressing Concept Maturity in the Early Formulation of Unmanned Spacecraft," *Proceedings of the 4th International Workshop on System and Concurrent Engineering for Space Applications, October 13-15, 2010*, Lausanne, Switzerland

Wessen, R., Borden, C., Ziemer, J., Kwok, J., "Space Mission Concept Development Using Concept Maturity Levels", *AIAA Space 2013 Proceedings*, September 10-12, 2013, San Diego, CA

Winner, R., Pennell, J., Bertrand, H., and Slusarczuk, M., *The Role Of Concurrent Engineering In Weapons System Acquisition*, Institute for Defense Analyses, IDA REPORT R-338, Dec 1988

Ziemer, J., Ervin, J., Lang, J., Exploring Mission Concepts with the JPL Innovation Foundry A-Team, *AIAA Space 2013 Proceedings,* September 10-12, 2013, San Diego, CA

National Research Council (NRC) of the National Academy of Sciences (NAS), The Planetary Decadal Survey 2013-2022, *Vision and Voyagers for Planetary Science in the Decade 2013-2022*, The National Academies Press: Washington, D.C., 2011. www.nap.edu

Section 7.3 Selecting Engineering Design Tools

NPR 2810.1, Security of Information Technology

NPR 7120.7, NASA Information Technology and Institutional Infrastructure Program and Project Management Requirements

Section 7.4 Environmental, Nuclear Safety, and Planetary Protection Policy Compliance

NPR 8000.4, Agency Risk Management Procedural Requirements

NPD 8020.7, Biological Contamination Control for Outbound and Inbound Planetary Spacecraft

NPI 8020.7, NASA Policy on Planetary Protection Requirements for Human Extraterrestrial Missions

NPR 8020.12, Planetary Protection Provisions for Robotic Extraterrestrial Missions

NPR 8580.1, NASA National Environmental Policy Act Management Requirements

NPR 8710.1, Emergency Preparedness Program

NPR 8715.2, NASA Emergency Preparedness Plan Procedural Requirements

NPR 8715.3, NASA General Safety Program Requirements

NASA Science Mission Directorate, *Risk Communication Plan for Planetary and Deep Space Missions*, 1999

National Environmental Policy Act of 1969 (NEPA). See 42 U.S.C. 4321-4347. https://ceq.doe.gov/welcome.html

Code of Federal Regulations (CFR), Title 14 – Aeronautics and Space, Part 1216.3 NASA Environmental Quality: Procedures for Implementing the National Environmental Policy Act (NEPA) (14 CFR 1216.3)

Code of Federal Regulations (CFR), Title 40 – Protection of Environment, Part 1508.27 Council on Environmental Quality: Terminology "significantly." (40 CFR 1508.27)

Executive Order (EO) 12114, *Environmental Effects Abroad of Major Federal Actions*. January 4, 1979

Presidential Directive/National Security Council Memorandum No. 25 (PD/NSC-25), "Scientific or Technological Experiments with Possible Large-Scale Adverse Environmental Effects and Launch of Nuclear Systems into Space," as amended May 8, 1996

United Nations, Office for Outer Space Affairs. *Treaty of Principles Governing the Activities of States in the Exploration and Use of Outer Space, Including the Moon and Other Celestial Bodies*. Known as the "Outer Space Treaty of 1967"

The Committee on Space Research (COSPAR) *Planetary Protection Policy*. March 24, 2005. http://w.astro.berkeley.edu/~kalas/ethics/documents/environment/COSPAR%20Planetary%20Protection%20Policy.pdf

Section 7.5 Use of Metric System

NPD 8010.2, Use of the SI (Metric) System of Measurement in NASA Programs

National Institute of Standards and Technology (NIST) Special Publication 330: *The International System of Units (SI)* Barry N. Taylor and Ambler Thompson, Editors, March 2008. The United States version of the English text of the eighth edition (2006) of the International Bureau of Weights and Measures publication Le Système International d' Unités (SI)

National Institute of Standards and Technology (NIST) Special Publication 811: *NIST Guide for the Use of the International System of Units (SI)* A. Thompson and B. N. Taylor, Editors. Created July 2, 2009; Last updated January 28, 2016

The Metric Conversion Act of 1975 (Public Law 94-168) amended by the Omnibus Trade and Competitiveness Act of 1988 (Public Law 100-418), the Savings in Construction Act of 1996 (Public Law 104-289), and the Department of Energy High-End Computing Revitalization Act of 2004 (Public Law 108-423). See 15 USC 205a et seq.

Executive Order (EO) 12770 *Metric Usage in Federal Government Programs*, July 25, 1991

Department of Defense (DOD) Office of the Under Secretary of Defense, Acquisition, Technology, & Logistics. SD-10. Defense Standardization Program: *Guide for Identification and Development of Metric Standards*. Washington, DC, April, 2010

Section 7.6 Systems Engineering on Multi-Level/Multi-Phase Programs

NPR 7123.1, NASA Systems Engineering Processes and Requirements

Section 7.7 Fault Management

NASA-HDBK-1002, *Fault Management (FM) Handbook*, Draft 2, April 2012

Jennions, Ian K. editor. *Integrated Vehicle Health Management (IVHM): Perspectives on an Emerging Field*. SAE International, Warrendale PA, 2011 IVHM Book, September 27, 2011

Jennions, Ian K. editor. *Integrated Vehicle Health Management (IVHM): Business Case Theory and Practice*. SAE International, Warrendale PA, 2012 IVHM Book, November 12, 2012

Jennions, Ian K. editor. *Integrated Vehicle Health Management (IVHM): The Technology*. SAE International, Warrendale PA, 2013 IVHM Book, September 5, 2013

Section 7.8 Technical Margins

NASA Cost Symposium 2014, NASA "Mass Growth Analysis - Spacecraft & Subsystems." LaRC, August 14th, 2014. Presenter: Vincent Larouche – Tecolote Research, also James K. Johnson, NASA HQ Study Point of Contact

APR 8070.2, EMI/EMC Class D Design and Environmental Test Requirements. NASA Ames Research Center (ARC)

NASA Goddard Space Flight Center, GSFC-STD-1000, *Rules for the Design, Development, Verification, and Operation of Flight Systems*. February 8, 2013

NASA Jet Propulsion Laboratory (JPL), JPL-D-17868 (REV.1), *JPL Guideline: Design, Verification/Validation and Operations Principles for Flight Systems*. February 16, 2001

Aerospace Conference 2007 IEEE Big Sky, MT 3-10 March 2007. NASA/Aerospace Corp. paper: "Using Historical NASA Cost and Schedule Growth to Set Future Program and Project Reserve Guidelines," by Emmons, D. L., R.E. Bitten, and C.W. Freaner. *IEEE Conference Publication* pages: 1-16, 2008. Also presented at the NASA Cost Symposium, Denver CO, July 17-19, 2007

Planetary Science Subcommittee, NASA Advisory Council, 23 June, 2008, NASA GSFC. NASA/Aerospace Corp. presentation; "An Assessment of the Inherent Optimism in Early Conceptual Designs and its Effect on Cost and Schedule Growth," by Freaner, Claude, Bob Bitten, Dave Bearden, and Debra Emmons

American Institute of Aeronautics and Astronautics (AIAA) S-120-2006, *Mass Properties Control for Space Systems*. Reston, VA 2006

American Institute of Aeronautics and Astronautics (AIAA) S-122-2007, *Electrical Power Systems for Unmanned Spacecraft*. Reston, VA 2007

Section 7.9 Human Systems Integration (HSI) in the SE Process

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7120.11, NASA Health and Medical Technical Authority (HMTA) Implementation

NPR 7123.1, NASA Systems Engineering Processes and Requirements

NPR 8705.2, Human-Rating Requirements for Space Systems

NPR 8900.1, NASA Health and Medical Requirements for Human Space Exploration

NASA-STD-3001, *Space Flight Human System Standard*. Volume 2: Human Factors, Habitability, and Environmental Health. Rev. A, February 10, 2015.

NASA Lyndon B. Johnson Space Center (JSC-65995), *Commercial Human Systems Integration Processes (CHSIP)*, May 2011.

NASA/SP-2010-3407, *Human Integration Design Handbook (HIDH)*

NASA/SP-2014-3705, *NASA Space Flight Program and Project Management Handbook*

NASA/SP-2015-3709 *Human Systems Integration Practitioners Guide*

NASA/TM-2008-215126/Volume II (NESC-RP-06-108/05-173-E/Part 2), Technical Memorandum: *Design Development Test and Evaluation (DDT&E) Considerations for Safe and Reliable Human-Rated Spacecraft Systems*. April 2008.Volume II: Technical Consultation Report. James Miller, Jay Leggett, and Julie Kramer-White, NASA Langley Research Center, Hampton VA, June 14, 2007.

NASA/TP-2014-218556, Technical Publication: *Human Integration Design Processes (HIDP).* NASA ISS Program, Lyndon B. Johnson Space Center, Houston TX, September 2014. http://ston.jsc.nasa.gov/collections/TRS/_techrep/TP-2014-218556.pdf

Federal Aviation Administration (FAA), HF-STD-001, *Human Factors Design Standard (HFDS).* Washington, DC, May 2003. Updated: May 03, 2012. hf.tc.faa.gov/hfds

Department of Defense (DOD) Defense Technical Information Center (DTIC). *Directory of Design Support Methods (DDSM)*. 2007. http://www.dtic.mil/dtic/tr/fulltext/u2/a437106.pdf

Section 8.1 Statistical Engineering as a Tool

American Society for Quality (ASQ), Statistics Division, Statistical Engineering, http://asq.org/statistics/quality-information/statistical-engineering

Hoerl, R.W. and R.S. Snee, *Statistical Thinking - Improving Business Performance*, John Wiley & Sons. 2012

NASA 2011 Statistical Engineering Symposium, Proceedings. http://engineering.larc.nasa.gov/2011_NASA_Statistical_Engineering_Symposium.html

Shaprio, J. (1994), "George H. Heilmeier," *IEEE Spectrum*, 31(6), pg. 56 – 59 http://ieeexplore.ieee.org/iel3/6/7047/00284787.pdf?arnumber=284787

Section 8.2 Model-Based Systems Engineering

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

Department of Defense (DOD) Architecture Framework (DODAF) Version 2.02 Change 1, January 2015 http://dodcio.defense.gov/Library/DoDArchitectureFramework.aspx

Architecture Analysis & Design Language (AADL):
https://wiki.sei.cmu.edu/aadl/index.php/Main_Page

Business Process Modeling Notation (BPMN) http://www.bpmn.org/

Friedenthal, Sanford, Alan Moore, and Rick Steiner. *A Practical Guide to SysML: Systems Modeling Language*, Morgan Kaufmann Publishers, Inc., July 2008

Institute of Electrical and Electronics Engineers (IEEE) STD 1076-2008 *IEEE Standard VHDL Language Reference Manual*, 03 February 2009

International Council on Systems Engineering (INCOSE). INCOSE-TP-2004-004-02, *Systems Engineering Vision 2020*, Version 2.03, September 2007
http://www.incose.org/ProductsPubs/pdf/SEVision2020_20071003_v2_03.pdf

International Organization for Standardization (ISO) 10303-AP233, *Application Protocol (AP) for Systems Engineering Data Exchange (AP-233)* Working Draft 2 published July 2006

International Organization for Standardization (ISO) ISO/TS 10303-433:2011 *Industrial automation systems and integration – Product data representation and exchange – Part 433: Application module: AP233 systems engineering*. ISO: Geneva, 2011

International Organization for Standardization (ISO). ISO/IEC/IEEE 42010:2011. *Systems and Software Engineering – Architecture Description.* (http://www.iso-architecture.org/ieee-1471/index.html) Geneva, 2011.

Knowledge Based Systems, Inc. (KBSI), *Integration Definition for functional modeling (IDEF0) ISF0 Function Modeling Method*, http://www.idef.com/idef0.htm,

Object Constraint Language (OCL) http://www.omg.org/spec/OCL/

Oliver, D., T. Kelliher, and J. Keegan. *Engineering Complex Systems with Models and Objects*. New York, NY, USA: McGraw-Hill. 1997.

Paredis, C., Y. Bernard, R. Burkhart, H.P. Koning, S. Friedenthal, P. Fritzon, N.F. Rouquette, W. Schamai. “Systems Modeling Language (SysML)-Modelica Transformation.” *INCOSE 2010*.

Query View Transformation (QVT) http://www.omg.org/spec/QVT/1.0/

SAE International, SAE Standard AS5506B: *Architecture Analysis & Design Language (AADL)* 2012-09-10

Systems Modeling Language (SysML) http://www.omgsysml.org/

Unified Modeling Language (UML) http://www.uml.org/

UPDM: Unified Profile for the (US) Department of Defense Architecture Framework (DoDAF) and the (UK) Ministry Of Defense Architecture Framework (MODAF) http://www.omg.org/spec/UPDM/

Web Ontology Language (OWL) http://www.w3.org/2001/sw/wiki/OWL

XMI: Extensible Markup Language (XML) Metadata Interchange (XMI) http://www.omg.org/spec/XMI/

XML: Extensible Markup Language (XML) http://www.w3.org/TR/REC-xml/

8.3 Concept Maturity Levels

Wessen, Randii R., Chester Borden, John Ziemer, and Johnny Kwok. "Space Mission Concept Development Using Concept Maturity Levels," Conference paper presented at the American Institute of Aeronautics and Astronautics (AIAA) Space 2013 Conference and Exposition; September 10-12, 2013; San Diego, CA. Published in the *AIAA Space 2013 Proceedings*.

Chattopadhyay, Debarati, Adam M. Ross, and Donna H. Rhodes, "A Method for Tradespace Exploration of Systems of Systems," presentation in Track 34-SSEE-3: Space Economic Cost Modeling, *AIAA Space 2009*, September 15, © 2009 Massachusetts Institute of Technology. SEARI: Systems Engineering Advancement Research Initiative, MIT. seari.mit.edu.

Appendix B Glossary

NPR 2210.1, Release of NASA Software

NPD 7120.4, NASA Engineering and Program/Project Management Policy

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

NPR 7150.2, NASA Software Engineering Requirements

NPR 8000.4, Agency Risk Management Procedural Requirements

NPR 8705.2, Human-Rating Requirements for Space Systems

NPR 8715.3, NASA General Safety Program Requirements

International Organization for Standardization (ISO). ISO/IEC/IEEE 42010:2011. *Systems and Software Engineering – Architecture Description*. Geneva: International Organization for Standardization, 2011. (http://www.iso-architecture.org/ieee-1471/index.html)

Avizienis, A., J.C. Laprie, B. Randell, C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *IEEE Transactions on Dependable and Secure Computing* 1 (1), 11-33, 2004

Appendix F: Functional, Timing, and State Analysis

NASA Reference Publication 1370, *Training Manual for Elements of Interface Definition and Control*. 1997

Defense Acquisition University. *Systems Engineering Fundamentals Guide*. Fort Belvoir, VA, 2001

Buede, Dennis. *The Engineering Design of Systems: Models and Methods*. New York: Wiley & Sons, 2000

Long, James E. *Relationships Between Common Graphical Representations in Systems Engineering*. Vienna, VA: Vitech Corporation, 2002

Sage, Andrew, and William Rouse. *The Handbook of Systems Engineering and Management*. New York: Wiley & Sons, 1999

Appendix G: Technology Assessment / Insertion

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7123.1, NASA Systems Engineering Processes and Requirements

Appendix H: Integration Plan Outline

Federal Highway Administration and CalTrans, *Systems Engineering Guidebook for ITS*, Version 2.0. Washington, DC: U.S. Department of Transportation, 2007

Appendix J: SEMP Content Outline

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

NPR 7123.1, Systems Engineering Processes and Requirements

Appendix K: Technical Plans

NPR 7120.5, NASA Space Flight Program and Project Management Requirements

Appendix M: CM Plan Outline

SAE International (SAE) / Electronic Industries Alliance (EIA) 649B-2011, *Configuration Management Standard (Aerospace Sector)* April 1, 2011

Appendix N: Guidance on Technical Peer Reviews/Inspections

NPR 7123.1, Systems Engineering Processes and Requirements

NPR 7150.2, NASA Software Engineering Requirements

NASA Langley Research Center (LARC), *Instructional Handbook for Formal Inspections*. http://sw-eng.larc.nasa.gov/files/2013/05/Instructional-Handbook-for-Formal-Inspections.pdf

Appendix P: SOW Review Checklist

NASA Langley Research Center (LaRC) Procedural Requirements (LPR) 5000.2 Procurement Initiator's Guide

NASA Langley Research Center (LaRC) *Guidance on System and Software Metrics for Performance-Based Contracting* sites-e.larc.nasa.gov/sweng/files/2013/05/Guidance_on_Metrics_for_PBC_R1V01.doc

Appendix R: HSI Plan Content Outline

NPR 7123.1, NASA Systems Engineering Processes and Requirements

NPR 8705.2, Human-Rating Requirements for Space Systems

NASA-STD-3001, *Space Flight Human-System Standard*, Volume 2: Human Factors, Habitability, and Environmental Health, Section 3.5 [V2 3005], "Human-Centered Design Process." February 10, 2015

Bibliography

The bibliography contains sources cited in sections of the document and additional sources for developing the material in the document.

AIAA	American Institute of Aeronautics and Astronautics
ANSI	American National Standards Institute
ASME	American Society of Mechanical Engineers
ASQ	American Society for Quality
CCSDS	Consultative Committee for Space Data Systems
CFR	(U.S.) Code of Federal Regulations
COSPAR	The Committee on Space Research
DOD	(U.S.) Department of Defense
EIA	Electronic Industries Alliance
GEIA	Government Electronics Information Technology Association
IEEE	Institute of Electrical and Electronics Engineers
INCOSE	International Council on Systems Engineering
ISO	International Organization for Standardization
NIST	National Institute of Standards and Technology
SAE	Society of Automotive Engineers
TOR	Technical Operating Report
U.S.C.	United States Code

A

Adams, R. J., et al. *Software Development Standard for Space Systems, Aerospace Corporation Report* No. TOR-2004(3909)3537, Revision B. March 11, 2005. Prepared for the U.S. Air Force.

AIAA G-118-2006e, *AIAA Guide for Managing the Use of Commercial Off the Shelf (COTS) Software Components for Mission-Critical Systems*, Reston, VA, 2006

AIAA S-120-2006, *Mass Properties Control for Space Systems*. Reston, VA, 2006

AIAA S-122-2007, *Electrical Power Systems for Unmanned Spacecraft*, Reston, VA, 2007

ANSI/AIAA G-043-1992, *Guide for the Preparation of Operational Concept Documents*, Washington, DC, 1992

ANSI/EIA-632, *Processes for Engineering a System*, Arlington, VA, 1999

ANSI/EIA-649, *National Consensus Standard for Configuration Management*, 1998-1999

ANSI/GEIA-649, *National Consensus Standard for Configuration Management*, National Defense Industrial Association (NDIA), Arlington, VA 1998

ANSI/EIA-748-C *Standard: Earned Value Management Systems*, March, 2013

ANSI/GEIA GEIA-859, *Data Management*, National Defense Industrial Association (NDIA), Arlington, VA 2004

ANSI/IEEE STD 1042. *IEEE Guide to Software Configuration Management*. Washington, DC, 1987

Architecture Analysis & Design Language (AADL): https://wiki.sei.cmu.edu/aadl/index.php/Main_Page

(The) Arms Export Control Act (AECA) of 1976, see 22 U.S.C. 2778

ASME Y14.24, *Types and Applications of Engineering Drawings*, New York, 1999

ASME Y14.100, *Engineering Drawing Practices*, New York, 2004

ASQ, Statistics Division, Statistical Engineering, http://asq.org/statistics/quality-information/statistical-engineering

Avizienis, A., J.C. Laprie, B. Randell, C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *IEEE Transactions on Dependable and Secure Computing* 1 (1), 11-33, 2004

B

Ball, Robert E. *The Fundamentals of Aircraft Combat Survivability Analysis and Design*. 2nd Edition, AIAA Education Series, 2003

Bayer, T.J., M. Bennett, C. L. Delp, D. Dvorak, J. S. Jenkins, and S. Mandutianu. "Update: Concept of Operations for Integrated Model-Centric Engineering at JPL," paper #1122, *IEEE Aerospace Conference 2011*

Blanchard, Benjamin S., *System Engineering Management*. 4th Edition, Hoboken, NJ: John Wiley & Sons, Inc., 2008

Blanchard, Benjamin S., and Wolter J. Fabrycky. *Systems Engineering and Analysis*, 5th Edition Prentice Hall International Series in Industrial & Systems Engineering; February 6, 2010

Brown, Barclay. "Model-based systems engineering: Revolution or Evolution," IBM Software, Thought Leadership White Paper, *IBM Rational*, December 2011

Brughelli, Kevin (Lockheed Martin), Deborah Carstens (Florida Institute of Technology), and Tim Barth (Kennedy Space Center), "Simulation Model Analysis Techniques," Lockheed Martin presentation to KSC, November 2003

Buede, Dennis. *The Engineering Design of Systems: Models and Methods*. New York: Wiley & Sons, 2000.

Business Process Modeling Notation (BPMN) http://www.bpmn.org/

C

CCSDS 311.0-M-1, *Reference Architecture for Space Data Systems,* Recommended Practice (Magenta), Sept 2008. http://public.ccsds.org/publications/MagentaBooks.aspx

CCSDS 901-0-G-1, *Space Communications Cross Support Architecture Description Document*, Informational Report (Green) Sept 2013. http://public.ccsds.org/publications/GreenBooks.aspx

Chapanis, A. "The Error-Provocative Situation: A Central Measurement Problem in Human Factors Engineering." In *The Measurement of Safety Performance*. Edited by W. E. Tarrants. New York: Garland STPM Press, 1980

Chattopadhyay, Debarati, Adam M. Ross, and Donna H. Rhodes, "A Method for Tradespace Exploration of Systems of Systems," presentation in Track 34-SSEE-3: Space Economic Cost Modeling, *AIAA Space 2009*, September 15, 2009. © 2009 Massachusetts Institute of Technology (MIT), SEARI: Systems Engineering Advancement Research Initiative, seari.mit.edu

Chung, Seung H., Todd J. Bayer, Bjorn Cole, Brian Cooke, Frank Dekens, Christopher Delp, Doris Lam. "Model-Based Systems Engineering Approach to Managing Mass Margin," in *Proceedings of the 5th International Workshop on Systems & Concurrent Engineering for Space Applications (SECESA),* Lisbon, Portugal, October, 2012

Clark, J.O. "System of Systems Engineering and Family of Systems Engineering From a Standards, V-Model, and Dual-V Model Perspective," *3rd Annual IEEE International Systems Conference*, Vancouver, Canada, March 23-26, 2009

Clemen, R., and T. Reilly. *Making Hard Decisions with DecisionTools Suite*. Pacific Grove, CA: Duxbury Resource Center, 2002

CFR, Title 14 – Aeronautics and Space, Part 1214 NASA Space Flight (14 CFR 1214)

CFR, Title 14 – Aeronautics and Space, Part 1216.3 NASA Environmental Quality: Procedures for Implementing the National Environmental Policy Act (NEPA) (14 CFR 1216.3)

CFR Title 22 – Foreign Relations, Parts 120-130 Department of State: International Traffic in Arms Regulations (ITAR) (22 CFR 120-130). Implements 22 U.S.C. 2778 of the Arms Export Control Act (AECA) of 1976 and Executive Order 13637, "Administration of Reformed Export Controls," March 8, 2013

CFR Title 37 – Patents, Trademarks, and Copyrights; Part 5 Secrecy of Certain Inventions and Licenses to Export and File Applications in Foreign Countries; Part 5.2 Secrecy Order. (37 CFR 5.2)

CFR Title 40 – Protection of Environment, Part 1508.27 Council on Environmental Quality: Terminology "significantly." (40 CFR 1508.27)

CFR Title 48 – Federal Acquisition Regulation (FAR) System, Part 1214 NASA Acquisition Planning: Acquisition of Commercial Items: Space Flight. (48 CFR 1214)

CFR Title 48 – Federal Acquisition Regulation (FAR) System, Part 46.103 Government Contract Quality Assurance: Contracting office responsibilities. (48 CFR 46.103)

CFR Title 48 – Federal Acquisition Regulation (FAR) System, Part 46.4 Government Contract Quality Assurance (48 CFR 46.4)

CFR Title 48 – Federal Acquisition Regulation (FAR) System, Part 46.407 Government Contract Quality Assurance: Nonconforming Supplies or Services (48 CFR 46.407)

COSPAR, *Planetary Protection Policy*. March 24, 2005. http://w.astro.berkeley.edu/~kalas/ethics/documents/environment/COSPAR%20Planetary%20Protection%20Policy.pdf

D

Deming, W. Edwards, see https://www.deming.org/

Dezfuli, H. "Role of System Safety in Risk-informed Decisionmaking." In *Proceedings, the NASA Risk Management Conference 2005*. Orlando, December 7, 2005

DOD Architecture Framework (DODAF) Version 2.02 Change 1, January 2015 http://dodcio.defense.gov/Library/DoDArchitectureFramework.aspx

DOD. *Defense Acquisition Guidebook (DAG).* 2014

DOD Defense Acquisition University (DAU). *Systems Engineering Fundamentals Guide*. Fort Belvoir, VA, 2001

DOD Defense Logistics Agency (DLA). *Cataloging Handbook, H4/H8 Series*. Washington, DC, February 2003

DOD Defense Technical Information Center (DTIC). *Directory of Design Support Methods (DDSM)*. 2007. http://www.dtic.mil/dtic/tr/fulltext/u2/a437106.pdf

DOD MIL-HDBK-727 (Validation Notice 1). *Military Handbook: Design Guidance for Producibility*, U.S. Army Research Laboratory, Weapons and Materials Research Directorate: Adelphi,MD, 1990

DOD. MIL-HDBK-965. *Acquisition Practices for Parts Management*. Washington, DC, September 26, 1996. Notice 1: October 2000

DOD. MIL-STD-881C. *Work Breakdown Structure (WBS) for Defense Materiel Items*. Washington, DC, October 3, 2011

DOD. MIL-STD-1472G, *DOD Design Criteria Standard: Human Engineering*. Washington, DC, January 11, 2012

DOD. MIL-STD-1540D, *Product Verification Requirements for Launch, Upper Stage, and Space Vehicles*. January 15, 1999

DOD. MIL-STD-46855A, *Human Engineering Requirements for Military Systems, Equipment, and Facilities*. May 24, 2011. Replacement for DOD HDBK 763 and DOD MIL-HDBK-46855A, which have been cancelled.

DOD Office of the Under Secretary of Defense, Acquisition, Technology, & Logistics. SD-10. *Defense Standardization Program: Guide for Identification and Development of Metric Standards*. Washington, DC, April, 2010

DOD Systems Management College. *Systems Engineering Fundamentals*. Defense Acquisition University Press: Fort Belvoir, VA 22060-5565, 2001 http://ocw.mit.edu/courses/aeronautics-and-astronautics/16-885j-aircraft-systems-engineering-fall-2005/readings/sefguide_01_01.pdf

Duren, R. et al., "Systems Engineering for the Kepler Mission: A Search for Terrestrial Planets," *IEEE Aerospace Conference*, 2006

E

Eggemeier, F. T., and G. F. Wilson. "Performance and Subjective Measures of Workload in Multitask Environments." In *Multiple-Task Performance*. Edited by D. Damos. London: Taylor and Francis, 1991

Endsley, M. R., and M. D. Rogers. "Situation Awareness Information Requirements Analysis for En Route Air Traffic Control." In *Proceedings of the Human Factors and Ergonomics Society 38th Annual Meeting*. Santa Monica: Human Factors and Ergonomics Society, 1994

Eslinger, Suellen. *Software Acquisition Best Practices for the Early Acquisition Phases*. El Segundo, CA: The Aerospace Corporation, 2004

Estefan, Jeff, *Survey of Model-Based Systems Engineering (MBSE) Methodologies*, Rev B, Section 3.2. NASA Jet Propulsion Laboratory (JPL), June 10, 2008. The document was originally authored as an internal JPL report, and then modified for public release and submitted to INCOSE to support the INCOSE MBSE Initiative.

Executive Order (EO) 12114, *Environmental Effects Abroad of Major Federal Actions*. January 4, 1979.

Executive Order (EO) 12770, *Metric Usage in Federal Government Programs*, July 25, 1991.

Executive Order (EO) 13637, *Administration of Reformed Export Controls*, March 8, 2013.

Extensible Markup Language (XML) http://www.w3.org/TR/REC-xml/

Extensible Markup Language (XML) Metadata Interchange (XMI) http://www.omg.org/spec/XMI/

F

Federal Acquisition Regulation (FAR). See: Code of Federal Regulations (CFR), Title 48.

Federal Aviation Administration (FAA), HF-STD-001, *Human Factors Design Standard (HFDS).* Washington, DC, May 2003. Updated: May 03, 2012. hf.tc.faa.gov/hfds

Federal Highway Administration, and CalTrans. *Systems Engineering Guidebook for ITS*, Version 2.0. Washington, DC: U.S. Department of Transportation, 2007.

Friedenthal, Sanford, Alan Moore, and Rick Steiner. *A Practical Guide to SysML: Systems Modeling Language*, Morgan Kaufmann Publishers, Inc., July 2008.

Fuld, R. B. "The Fiction of Function Allocation." *Ergonomics in Design* (January 1993): 20–24.

G

Garlan, D., W. Reinholtz, B. Schmerl, N. Sherman, T. Tseng. "Bridging the Gap between Systems Design and Space Systems Software," *Proceedings of the 29th IEEE/NASA Software Engineering Workshop*, 6-7 April 2005, Greenbelt, MD, USA

Glass, J. T., V. Zaloom, and D. Gates. "A Micro-Computer-Aided Link Analysis Tool." *Computers in Industry* 16, (1991): 179–87

Gopher, D., and E. Donchin. "Workload: An Examination of the Concept." In *Handbook of Perception and Human Performance: Vol. II. Cognitive Processes and Performance*. Edited by K. R. Boff, L. Kaufman, and J. P. Thomas. New York: John Wiley & Sons, 1986

Griffin, Michael D., NASA Administrator. "System Engineering and the Two Cultures of Engineering." Boeing Lecture, Purdue University, March 28, 2007

H

Hart, S. G., and C. D. Wickens. "Workload Assessment and Prediction." In *MANPRINT: An Approach to Systems Integration*. Edited by H. R. Booher. New York: Van Nostrand Reinhold, 1990

Hoerl, R.W. and R.S. Snee, *Statistical Thinking - Improving Business Performance*, John Wiley & Sons. 2012

Hoffmann, Hans-Peter, "Harmony-SE/SysML Deskbook: Model-Based Systems Engineering with Rhapsody," Rev. 1.51, Telelogic/I-Logix white paper, Telelogic AB, May 24, 2006

Hofmann, Hubert F., Kathryn M. Dodson, Gowri S. Ramani, and Deborah K. Yedlin. *Adapting CMMI® for Acquisition Organizations: A Preliminary Report*, CMU/ SEI-2006-SR-005. Pittsburgh: Software Engineering Institute, Carnegie Mellon University, 2006, pp. 338–40

Huey, B. M., and C. D. Wickens, eds. *Workload Transition*. Washington, DC: National Academy Press, 1993

IEEE STD 610.12-1990. *IEEE Standard Glossary of Software Engineering Terminology*. 1999, superceded by ISO/IEC/IEEE 24765:2010, *Systems and Software Engineering – Vocabulary*. Washington, DC, 2010

IEEE STD 828. *IEEE Standard for Software Configuration Management Plans*. Washington, DC, 1998

IEEE STD 1076-2008 *IEEE Standard VHDL Language Reference Manual*, 03 February 2009

IEEE STD 1220-2005. *IEEE Standard for Application and Management of the Systems Engineering Process*, Washington, DC, 2005

IEEE Standard12207.1, *EIA Guide for Information Technology Software Life Cycle Processes—Life Cycle Data*, Washington, DC, 1997

INCOSE. *Systems Engineering Handbook*, Version 3.2.2. Seattle, 2011

INCOSE-TP-2003-002-04, *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*, Version 4, Edited by Walden, David D., et al., 2015

INCOSE-TP-2003-020-01, *Technical Measurement*, Version 1.0, 27 December 2005. Prepared by Garry J. Roedler (Lockheed Martin) and Cheryl Jones (U.S. Army).

INCOSE-TP-2004-004-02, *Systems Engineering Vision 2020*, Version 2.03, September 2007, http://www.incose.org/ProductsPubs/pdf/SEVision2020_20071003_v2_03.pdf

INCOSE-TP-2005-001-03, *Systems Engineering Leading Indicators Guide*, Version 2.0, January 29, 2010; available at http://seari.mit.edu/documents/SELI-Guide-Rev2.pdf. Edited by Garry J. Roedler and Howard Schimmoller (Lockheed Martin), Cheryl Jones (U.S. Army), and Donna H. Rhodes (Massachusetts Institute of Technology)

ISO 9000:2015, *Quality management systems - Fundamentals and vocabulary*. Geneva: International Organization for Standardization, 2015

ISO 9001:2015 *Quality Management Systems (QMS).* Geneva: International Organization for Standardization, September 2015

ISO 9100/AS9100, *Quality Systems Aerospace—Model for Quality Assurance in Design, Development, Production, Installation, and Servicing*. Geneva: International Organization for Standardization, 1999

ISO 10007: 1995(E). *Quality Management—Guidelines for Configuration Management*, Geneva: International Organization for Standardization, 1995

ISO 10303-AP233, *Application Protocol (AP) for Systems Engineering Data Exchange (AP-233)* Working Draft 2 published July 2006

ISO/TS 10303-433:2011 *Industrial automation systems and integration – Product data representation and exchange – Part 433: Application module: AP233 systems engineering*. ISO: Geneva, 2011

ISO/IEC 10746-1 to 10746-4, *ITU-T Specifications X.901 to x.904, Reference Model of Open distributed Processing (RM-ODP),* Geneva: International Organization for Standardization, 1998. www.rm-odp.net

ISO 13374-1, *Condition monitoring and diagnostics of machines – Data processing, communication and presentation - Part 1: General guidelines*. Geneva: International Organization for Standardization, 2002

ISO/IEC 15288:2002. *Systems Engineering—System Life Cycle Processes*. Geneva: International Organization for Standardization, 2002

ISO/TR 15846. *Information Technology—Software Life Cycle Processes Configuration Management*, Geneva: International Organization for Standardization, 1998

ISO/IEC TR 19760:2003. Systems Engineering—A Guide for the Application of ISO/IEC 15288. Geneva: International Organization for Standardization, 2003

ISO/IEC/IEEE 24765:2010, Systems and Software Engineering – Vocabulary. Geneva: International Organization for Standardization, 2010

ISO/IEC/IEEE 42010:2011. Systems and Software Engineering – Architecture Description. Geneva: International Organization for Standardization, 2011 http://www.iso-architecture.org/ieee-1471/index.html

(The) Invention Secrecy Act of 1951, see 35 U.S.C. §181-§188. Secrecy of Certain Inventions and Filing Applications in Foreign Country; §181 - Secrecy of Certain Inventions and Withholding of Patent

J

Joint (cost and schedule) Confidence Level (JCL). Frequently asked questions (FAQs) can be found at: http://www.nasa.gov/pdf/394931main_JCL_FAQ_10_12_09.pdf

Jennions, Ian K. editor. *Integrated Vehicle Health Management (IVHM): Perspectives on an Emerging Field*. SAE International, Warrendale PA (IVHM Book) September 27, 2011

Jennions, Ian K. editor. *Integrated Vehicle Health Management (IVHM): Business Case Theory and Practice*. SAE International, Warrendale PA (IVHM Book) November 12, 2012

Jennions, Ian K. editor. *Integrated Vehicle Health Management (IVHM): The Technology*. SAE International, Warrendale PA (IVHM Book) September 5, 2013

Johnson, Stephen B. et al., editors. *System Health Management with Aerospace Applications*. John Wiley & Sons, Ltd, West Sussex, UK, 2011

Jones, E. R., R. T. Hennessy, and S. Deutsch, eds. *Human Factors Aspects of Simulation*. Washington, DC: National Academy Press, 1985

K

Kaplan, S., and B. John Garrick. "On the Quantitative Definition of Risk." *Risk Analysis* 1(1). 1981

Karpati, G., Martin, J., Steiner, M., Reinhardt, K., "The Integrated Mission Design Center (IMDC) at NASA Goddard Space Flight Center," *IEEE Aerospace Conference 2003 Proceedings*, Volume 8, Page(s): 8_3657 - 8_3667, 2003

Keeney, Ralph L. *Value-Focused Thinking: A Path to Creative Decisionmaking*. Cambridge, MA: Harvard University Press, 1992

Keeney, Ralph L., and Timothy L. McDaniels. "A Framework to Guide Thinking and Analysis Regarding Climate Change Policies." *Risk Analysis* 21(6): 989–1000. 2001

Keeney, Ralph L., and Howard Raiffa. *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. Cambridge, UK: Cambridge University Press, 1993

Kirwin, B., and L. K. Ainsworth. *A Guide to Task Analysis*. London: Taylor and Francis, 1992

Kluger, Jeffrey with Dan Cray, "Management Tips from the Real Rocket Scientists," *Time Magazine*, November 2005

Knowledge Based Systems, Inc. (KBSI), *Integration Definition for functional modeling (IDEF0) ISF0 Function Modeling Method*, found at http://www.idef.com/idef0.htm

Kruchten, Philippe B. *The Rational Unified Process: An Introduction*, Third Edition, Addison-Wesley Professional: Reading, MA, 2003

Kruchten, Philippe B. "A 4+1 view model of software architecture," *IEEE Software Magazine* 12(6) (November 1995), 42-50

Kurke, M. I. "Operational Sequence Diagrams in System Design." *Human Factors* 3: 66–73. 1961

L

Larson, Wiley J.et al.. *Applied Space Systems Engineering: A Practical Approach to Achieving Technical Baselines*. 2nd Edition, Boston, MA: McGraw-Hill Learning Solutions, CEI Publications, 2009

Long, James E., *Relationships Between Common Graphical Representations in Systems Engineering*. Vienna, VA: Vitech Corporation, 2002

Long, James E., "Systems Engineering (SE) 101," *CORE®: Product & Process Engineering Solutions*, Vitech training materials. Vienna, VA: Vitech Corporation, 2000

M

Maier, M.W. “Architecting Principles for Systems-of-Systems,” *Systems Engineering* 1(1998), 267-284, John Wiley & Sons, Inc

Maier, M.W., D. Emery, and R. Hillard, “ANSI/IEEE 1471 and Systems Engineering,” *Systems Engineering* 7 (2004), 257-270, Wiley InterScience, www.interscience.wiley.com

Maier, M.W. “System and Software Architecture Reconciliation,” *Systems Engineering* 9 (2006), 146-159, Wiley InterScience, www.interscience.wiley.com

Maier, M.W., and E. Rechtin, *The Art of Systems Architecting*, 3rd Edition, CRC Press, Boca Raton, FL, 2009

Martin, James N., *Processes for Engineering a System: An Overview of the ANSI/GEIA EIA-632 Standard and Its Heritage*. New York: Wiley & Sons, 2000

Martin, James N., *Systems Engineering Guidebook: A Process for Developing Systems and Products*. Boca Raton: CRC Press, 1996.

Mathworks: Matlab http://www.mathworks.com/

McGuire, M., Oleson, S., Babula, M., and Sarver-Verhey, T., “Concurrent Mission and Systems Design at NASA Glenn Research Center: The origins of the COMPASS Team,” *AIAA Space 2011 Proceedings*, September 27-29, 2011, Long Beach, CA

Meister, David, *Behavioral Analysis and Measurement Methods*. New York: John Wiley & Sons, 1985

Meister, David, *Human Factors: Theory and Practice*. New York: John Wiley & Sons, 1971

(The) Metric Conversion Act of 1975 (Public Law 94-168) amended by the Omnibus Trade and Competitiveness Act of 1988 (Public Law 100-418), the Savings in Construction Act of 1996 (Public Law 104-289), and the Department of Energy High-End Computing Revitalization Act of 2004 (Public Law 108-423). See 15 U.S.C. §205a et seq.

Miao, Y., and J. M. Haake. “Supporting Concurrent Design by Integrating Information Sharing and Activity Synchronization.” In *Proceedings of the 5th ISPE International Conference on Concurrent Engineering Research and Applications (CE98)*. Tokyo, 1998, pp. 165–74

The Mitre Corporation, *Common Risks and Risk Mitigation Actions for a COTS-based System*. McLean, VA. www2.mitre.org/.../files/CommonRisksCOTS.doc (no date)

MODAF http://www.modaf.com/

Moeller, Robert C., Chester Borden, Thomas Spilker, William Smythe, Robert Lock , “Space Missions Trade Space Generation and Assessment using the JPL Rapid Mission Architecture (RMA) Team Approach,” *IEEE Aerospace Conference*, Big Sky, Montana, March 2011

Morgan, M. Granger, and M. Henrion, *Uncertainty: A Guide to Dealing with Uncertainty in Quantitative Risk and Policy Analysis*. Cambridge, UK: Cambridge University Press, 1990

M. Moshir, et al., "Systems engineering and application of system performance modeling in SIM Lite mission," *Proceedings. SPIE* 7734, 2010

Mulqueen, J.; R. Hopkins; D. Jones, "The MSFC Collaborative Engineering Process for Preliminary Design and Concept Definition Studies." 2012 http://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20120001572.pdf

NASA Publications

NASA Federal Acquisition Regulation (FAR) Supplement (NFS) 1834.201, Earned Value Management System Policy

NASA Form (NF) 1686, NASA Scientific and Technical Document Availability Authorization (DAA) for Administratively Controlled Information

Reports

NASA Chief Engineer and the NASA Integrated Action Team (NIAT) report, "Enhancing Mission Success -- A Framework for the Future," December 21, 2000. Authors: McBrayer, Robert O and Thomas, Dale, NASA Marshall Space Flight Center, Huntsville, AL United States

NASA. *Columbia Accident Investigation Board (CAIB) Report*, 6 volumes: Aug. 26, Oct. 2003. http://www.nasa.gov/columbia/caib/html/report.html

NASA. *NOAA N-Prime Mishap Investigation Final Report*, Sept. 13, 2004. http://www.nasa.gov/pdf/65776main_noaa_np_mishap.pdf

NASA. Diaz Report, *A Renewed Commitment to Excellence: An Assessment of the NASA Agency-wide Applicability of the Columbia Accident Investigation Board Report*, January 30, 2004. Mr. Al Diaz, Director, Goddard Space Flight Center, and team

NASA JPL D-71990, *Europa Study 2012 Full Report*. May 1 2012, publicly available here: http://solarsystem.nasa.gov/europa/2012study.cfm

NASA Office of Inspector General. *Final Memorandum on NASA's Acquisition Approach Regarding Requirements for Certain Software Engineering Tools to Support NASA Programs*, Assignment No. S06012. Washington, DC, 2006

NASA Office of Inspector General. *Performance-Based Contracting*_ https://oig.nasa.gov/august/report/FY06/s06012

Specialty Web Sites

NASA Engineering Network (NEN) Systems Engineering Community of Practice (SECoP) located at https://nen.nasa.gov/web/se

NASA Engineering Network (NEN) Systems Engineering Community of Practice (SECoP) under Tools and Methods at https://nen.nasa.gov/web/se/tools/ and then NASA Tools & Methods

NASA Engineering Network (NEN) V&V Community of Practice, located at https://nen.nasa.gov/web/se

NASA Engineering Network (NEN) EVM Community of Practice, https://nen.nasa.gov/web/pm/evm

NASA EVM website http://evm.nasa.gov/index.html

NASA Procurement Library found at http://www.hq.nasa.gov/office/procurement/

Conference Publications

NASA 2011 Statistical Engineering Symposium, Proceedings. http://engineering.larc.nasa.gov/2011_NASA_Statistical_Engineering_Symposium.html

Aerospace Conference, 2007 IEEE Big Sky, MT 3-10 March 2007. NASA/Aerospace Corp. paper: "Using Historical NASA Cost and Schedule Growth to Set Future Program and Project Reserve Guidelines," by Emmons, D. L., R.E. Bitten, and C.W. Freaner. IEEE Conference Publication pages: 1-16, 2008. Also presented at the NASA Cost Symposium, Denver CO, July 17-19, 2007

NASA Cost Symposium 2014, NASA "Mass Growth Analysis - Spacecraft & Subsystems." LaRC, August 14th, 2014. Presenter: Vincent Larouche – Tecolote Research, also James K. Johnson, NASA HQ Study Point of Contact

Planetary Science Subcommittee, NASA Advisory Council, 23 June, 2008, NASA GSFC. NASA/Aerospace Corp. presentation; "An Assessment of the Inherent Optimism in Early Conceptual Designs and its Effect on Cost and Schedule Growth," by Freaner, Claude, Bob Bitten, Dave Bearden, and Debra Emmons

Technical Documents

NASA Office of Chief Information Officer (OCIO). *Information Technology Systems Engineering Handbook* Version 2.0

NASA Science Mission Directorate, *Risk Communication Plan for Planetary and Deep Space Missions*, 1999

NASA PD-EC-1243, *Preferred Reliability Practices for Fault Protection*, October 1995

NASA-CR-192656, *Contractor Report: Research and technology goals and objectives for Integrated Vehicle Health Management (IVHM).* October 10, 1992

NASA Jet Propulsion Laboratory (JPL), JPL-D-17868 (REV.1), *JPL Guideline: Design, Verification/Validation and Operations Principles for Flight Systems*. February 16, 2001

NASA Lyndon B. Johnson Space Center (JSC-65995), *Commercial Human Systems Integration Processes (CHSIP)*, May 2011

NASA/TP-2014-218556, *Technical Publication: Human Integration Design Processes (HIDP).* NASA ISS Program, Lyndon B. Johnson Space Center, Houston TX, September 2014. http://ston.jsc.nasa.gov/collections/TRS/_techrep/TP-2014-218556.pdf

NASA Lyndon B. Johnson Space Center (JSC-60576), *National Space Transportation System (NSTS), Space Shuttle Program, Transition Management Plan*, May 9, 2007

NASA Langley Research Center (LARC) *Guidance on System and Software Metrics for Performance-Based Contracting*. 2013 sites-e.larc.nasa.gov/sweng/files/2013/05/Guidance_on_Metrics_for_PBC_R1V01.doc

NASA Langley Research Center (LARC), *Instructional Handbook for Formal Inspections*. 2013 http://sw-eng.larc.nasa.gov/files/2013/05/Instructional-Handbook-for-Formal-Inspections.pdf

NASA/TM-2008-215126/Volume II (NESC-RP-06-108/05-173-E/Part 2), *Technical Memorandum: Design Development Test and Evaluation (DDT&E) Considerations for Safe and Reliable Human-Rated Spacecraft Systems*. April 2008.Volume II: *Technical Consultation Report*. James Miller, Jay Leggett, and Julie Kramer-White, NASA Langley Research Center, Hampton VA, June 14, 2007

NASA Reference Publication 1370. *Training Manual for Elements of Interface Definition and Control.* Vincent R. Lalli, Robert E. Kastner, and Henry N. Hartt. NASA Lewis Research Center, Cleveland OH, January 1997

NASA. *Systems Engineering Leading Indicators Guide*, http://seari.mit.edu/

NASA *Cost Estimating Handbook (CEH)*, Version 4, February 2015

NASA *Financial Management Requirements (FMR)* Volume 4

Special Publications

NASA/SP-2010-576 *NASA Risk-Informed Decision Making Handbook*

NASA/SP-2012-599, *NASA's Earned Value Management (EVM) Implementation Handbook*

NASA/SP-2010-3403, *NASA Schedule Management Handbook*

NASA/SP-2010-3404, *NASA Work Breakdown Structure Handbook*

NASA/SP-2010-3406, *Integrated Baseline Review (IBR) Handbook*

NASA/SP-2010-3407, *Human Integration Design Handbook (HIDH)*

NASA/SP-2011-3421, *Probabilistic Risk Assessment Procedures Guide for NASA Managers and Practitioners*

NASA/SP-2011-3422, *NASA Risk Management Handbook*

NASA/SP-2013-3704, *Earned Value Management (EVM) System Description*

NASA/SP-2014-3705, *NASA Space Flight Program and Project Management Handbook*

NASA/SP-2015-3709, *Human Systems Integration Practitioners Guide*

Handbooks and Standards

NASA-HDBK-1002, *Fault Management (FM) Handbook*, Draft 2, April 2012

NASA-HDBK-2203, *NASA Software Engineering Handbook*, February 28, 2013

NASA Safety Standard (NSS) 1740.14, *Guidelines and Assessment Procedures for Limiting Orbital Debris*. Washington, DC, 1995
http://www.hq.nasa.gov/office/codeq/doctree/174014.htm
NASA-STD 8719.14 should be used in place of NSS 1740.14 to implement NPR 8715.6. See NPR 8715.6 for restrictions on the use of NSS 1740.14.

NASA GSFC-STD-1000, *Rules for the Design, Development, Verification, and Operation of Flight Systems*. NASA Goddard Space Flight Center, February 8, 2013

NASA-STD-3001, *Space Flight Human System Standard*. Volume 1: *Crew Health*. Rev. A, July 30, 2014

NASA-STD-3001, *Space Flight Human System Standard*. Volume 2: *Human Factors, Habitability, and Environmental Health*. Rev. A, February 10, 2015

NASA GSFC-STD-7000, *Goddard Technical Standard: General Environmental Verification Standard (GEVS) for GSFC Flight Programs and Projects*. Goddard Space Flight Center, April 2005

NASA KSC-NE-9439 *Kennedy Space Center Design Engineering Handbook, Best Practices for Design and Development of Ground Systems*. Kennedy Space Center, November 20 2009

NASA-STD-7009, *Standard for Models and Simulations*. Washington, DC, October 18, 2013

NASA-STD-8719.13, *Software Safety Standard*, Rev C. Washington, DC, May 7, 2013

NASA-STD-8719.14, *Handbook for Limiting Orbital Debris*. Rev A with Change 1. December 8, 2011

NASA-STD-8729.1, *Planning, Developing, and Maintaining an Effective Reliability and Maintainability (R&M) Program*. Washington, DC, December 1, 1998

Policy Directives

NPD 1001.0, 2014 NASA Strategic Plan

NID 1600.55, Sensitive But Unclassified (SBU) Controlled Information

NPD 2820.1, NASA Software Policy

NPD 7120.4, NASA Engineering and Program/Project Management Policy

NPD 7120.6, Knowledge Policy on Programs and Projects

NPD 8010.2, Use of the SI (Metric) System of Measurement in NASA Programs

NPD 8010.3, Notification of Intent to Decommission or Terminate Operating Space Systems and Terminate Missions

NPD 8020.7, Biological Contamination Control for Outbound and Inbound Planetary Spacecraft

NPD 8730.5, NASA Quality Assurance Program Policy

Procedural Requirements

NPR 1080.1, Requirements for the Conduct of NASA Research and Technology (R&T)

NPR 1441.1, NASA Records Management Program Requirements

NPR 1600.1, NASA Security Program Procedural Requirements

NPR 2210.1, Release of NASA Software

NPR 2810.1, Security of Information Technology

LPR 5000.2, Procurement Initiator's Guide. NASA Langley Research Center (LARC)

JPR 7120.3, Project Management: Systems Engineering & Project Control Processes and Requirements. NASA Lyndon B. Johnson Space Center (JSC)

NPR 7120.5, NASA Space Flight Program and Project Management Processes and Requirements

NPR 7120.7, NASA Information Technology and Institutional Infrastructure Program and Project Management Requirements

NPR 7120.8, NASA Research and Technology Program and Project Management Requirements

NPR 7120.10, Technical Standards for NASA Programs and Projects

NPR 7120.11, NASA Health and Medical Technical Authority (HMTA) Implementation

NPR 7123.1, Systems Engineering Processes and Requirements

NPR 7150.2, NASA Software Engineering Requirements

NPR 8000.4, Risk Management Procedural Requirements

NPI 8020.7, NASA Policy on Planetary Protection Requirements for Human Extraterrestrial Missions

NPR 8020.12, Planetary Protection Provisions for Robotic Extraterrestrial Missions

APR 8070.2, EMI/EMC Class D Design and Environmental Test Requirements. NASA Ames Research Center (ARC)

NPR 8580.1, Implementing the National Environmental Policy Act and Executive Order 12114

NPR 8705.2, Human-Rating Requirements for Space Systems

NPR 8705.3, Probabilistic Risk Assessment Procedures Guide for NASA Managers and Practitioners

NPR 8705.4, Risk Classification for NASA Payloads

NPR 8705.5, Probabilistic Risk Assessment (PRA) Procedures for NASA Programs and Projects

NPR 8705.6, Safety and Mission Assurance (SMA) Audits, Reviews, and Assessments

NPR 8710.1, Emergency Preparedness Program

NPR 8715.2, NASA Emergency Preparedness Plan Procedural Requirements

NPR 8715.3, NASA General Safety Program Requirements

NPR 8715.6, NASA Procedural Requirements for Limiting Orbital Debris

NPR 8735.2, Management of Government Quality Assurance Functions for NASA Contracts

NPR 8900.1, NASA Health and Medical Requirements for Human Space Exploration

Work Instructions

MSFC NASA MWI 8060.1, Off-the-Shelf Hardware Utilization in Flight Hardware Development. NASA Marshall Space Flight Center.

JSC Work Instruction EA-WI-016, Off-the-Shelf Hardware Utilization in Flight Hardware Development. NASA Lyndon B. Johnson Space Center.

Acquisition Documents

NASA. *The SEB Source Evaluation Process*. Washington, DC, 2001

NASA. *Solicitation to Contract Award*. Washington, DC, NASA Procurement Library, 2007

NASA. *Statement of Work Checklist*. Washington, DC. See: Appendix P in this guide.

N

(The) National Environmental Policy Act of 1969 (NEPA). See 42 U.S.C. 4321-4347. https://ceq.doe.gov/welcome.html

National Research Council (NRC) of the National Academy of Sciences (NAS), The Planetary Decadal Survey 2013-2022, *Vision and Voyagers for Planetary Science in the Decade 2013-2022*, The National Academies Press: Washington, D.C., 2011. www.nap.edu

NIST Special Publication 330: *The International System of Units (SI)* Barry N. Taylor and Ambler Thompson, Editors, March 2008. The United States version of the English text of the eighth edition (2006) of the International Bureau of Weights and Measures publication *Le Système International d' Unités (SI)*

NIST Special Publication 811: *NIST Guide for the Use of the International System of Units (SI)* A. Thompson and B. N. Taylor, Editors. Created July 2, 2009; Last updated January 28, 2016

NIST, Federal Information Processing Standard Publication (FIPS PUB) 199, *Standards for Security Categorization of Federal Information and Information Systems,* February 2004

O

Oberto, R.E., Nilsen, E., Cohen, R., Wheeler, R., DeFlorio, P., and Borden, C., "The NASA Exploration Design Team; Blueprint for a New Design Paradigm", *2005 IEEE Aerospace Conference*, Big Sky, Montana, March 2005

Object Constraint Language (OCL) http://www.omg.org/spec/OCL/

Office of Management and Budget (OMB) Circular A-94, *Guidelines and Discount Rates for Benefit-Cost Analysis of Federal Programs*, October 29, 1992

Oliver, D., T. Kelliher, and J. Keegan, *Engineering Complex Systems with Models and Objects*, New York, NY, USA: McGraw-Hill 1997

OOSEM Working Group, *Object-Oriented Systems Engineering Method (OOSEM) Tutorial*, Version 03.00, Lockheed Martin Corporation and INCOSE, October 2008

OWL, Web Ontology Language (OWL) http://www.w3.org/2001/sw/wiki/OWL

P

Paredis, C., Y. Bernard, R. Burkhart, H.P. Koning, S. Friedenthal, P. Fritzon, N.F. Rouquette, W. Schamai. "Systems Modeling Language (SysML)-Modelica Transformation." *INCOSE 2010*

Pennell, J. and Winner, R., "Concurrent Engineering: Practices and Prospects", Global Telecommunications Conference, *GLOBECOM '89*, 1989

Presidential Directive/National Security Council Memorandum No. 25 (PD/NSC-25), "Scientific or Technological Experiments with Possible Large-Scale Adverse Environmental Effects and Launch of Nuclear Systems into Space," as amended May 8, 1996

Presidential Policy Directive PPD-4 (2010), *National Space Policy*

Presidential Policy Directive PPD-21 (2013), *Critical Infrastructure Security and Resilience*

Price, H. E. "The Allocation of Functions in Systems." *Human Factors* 27: 33–45. 1985

The Project Management Institute® (PMI). *Practice Standards for Work Breakdown Structures*. Newtown Square, PA, 2001

Q

Query View Transformation (QVT) http://www.omg.org/spec/QVT/1.0/

R

Rasmussen, Robert. "Session 1: Overview of State Analysis," (internal document), *State Analysis Lite Course*, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, 2005

R. Rasmussen, B. Muirhead, *Abridged Edition: A Case for Model-Based Architecting in NASA*, California Institute of Technology, August 2012

Rechtin, Eberhardt. *Systems Architecting of Organizations: Why Eagles Can't Swim*. Boca Raton: CRC Press, 2000

S

Saaty, Thomas L. *The Analytic Hierarchy Process*. New York: McGraw-Hill, 1980

SAE Standard AS5506B, *Architecture Analysis & Design Language (AADL)*, SAE International, September 10, 2012

SAE International and the European Association of Aerospace Industries (EAAI) AS9100C, *Quality Management Systems (QMS) - Requirements for Aviation, Space, and Defense Organizations* Revision C, January 15, 2009

SAE International / Electronic Industries Alliance (EIA) 649B-2011, *Configuration Management Standard (Aerospace Sector)*, April 1, 2011

Sage, Andrew, and William Rouse. *The Handbook of Systems Engineering and Management*, New York: Wiley & Sons, 1999

Shafer, J. B. "Practical Workload Assessment in the Development Process." In *Proceedings of the Human Factors Society 31st Annual Meeting*, Santa Monica: Human Factors Society, 1987

Shames, P., and J. Skipper. "Toward a Framework for Modeling Space Systems Architectures," *SpaceOps 2006 Conference*, AIAA 2006-5581, 2006

Shaprio, J., "George H. Heilmeier," *IEEE Spectrum*, 31(6), 1994, pg. 56 – 59 http://ieeexplore.ieee.org/iel3/6/7047/00284787.pdf?arnumber=284787

Software Engineering Institute (SEI). *A Framework for Software Product Line Practice*, Version 5.0. Carnegie Mellon University, www.sei.cmu.edu/productlines/frame_report/arch_def.htm

Stamelatos, M., H. Dezfuli, and G. Apostolakis. "A Proposed Risk-Informed Decision making Framework for NASA." In *Proceedings of the 8th International Conference on Probabilistic Safety Assessment and Management*. New Orleans, LA, May 14–18, 2006

Stern, Paul C., and Harvey V. Fineberg, eds. *Understanding Risk: Informing Decisions in a Democratic Society*. Washington, DC: National Academies Press, 1996

Systems Modeling Language (SysML) http://www.omgsysml.org/

T

Taylor, Barry. *Guide for the Use of the International System of Units (SI),* Special Publication 811. Gaithersburg, MD: NIST, Physics Laboratory, 2007

U

Unified Modeling Language (UML) http://www.uml.org/

UPDM: Unified Profile for the (US) Department of Defense Architecture Framework (DoDAF) and the (UK) Ministry Of Defense Architecture Framework (MODAF) http://www.omg.org/spec/UPDM/

U.S. Air Force. *SMC Systems Engineering Primer and Handbook*, 3rd ed. Los Angeles: Space and Missile Systems Center, 2005

U. S. Chemical Safety Board (CSB) case study reports on mishaps found at: http://www.csb.gov/

U.S. Navy. Naval Air Systems Command, *Systems Engineering Guide*: 2003 (based on requirements of ANSI/EIA 632:1998). Patuxent River, MD, 2003

U.S. Nuclear Regulatory Commission. SECY-98-144, *White Paper on Risk-Informed and Performance-Based Regulation*,Washington, DC, 1998

U.S. Nuclear Regulatory Commission. NUREG-0700, *Human-System Interface Design Review Guidelines*, Rev.2. Washington, DC, Office of Nuclear Regulatory Research, 2002

United Nations, Office for Outer Space Affairs. *Treaty of Principles Governing the Activities of States in the Exploration and Use of Outer Space, Including the Moon and Other Celestial Bodies*. Known as the "Outer Space Treaty of 1967"

W

Wall, S., "Use of Concurrent Engineering in Space Mission Design," *Proceedings of EuSEC 2000*, Munich, Germany, September 2000

Warfield, K., “Addressing Concept Maturity in the Early Formulation of Unmanned Spacecraft,” *Proceedings of the 4th International Workshop on System and Concurrent Engineering for Space Applications*, October 13-15, 2010, Lausanne, Switzerland

Web Ontology Language (OWL) http://www.w3.org/2001/sw/wiki/OWL

Wessen, Randii R., Chester Borden, John Ziemer, and Johnny Kwok. “Space Mission Concept Development Using Concept Maturity Levels,” Conference paper presented at the American Institute of Aeronautics and Astronautics (AIAA) Space 2013 Conference and Exposition; September 10-12, 2013; San Diego, CA. Published in the *AIAA Space 2013 Proceedings*

Winner, R., Pennell, J., Bertrand, H., and Slusarczuk, M., *The Role Of Concurrent Engineering In Weapons System Acquisition*, Institute of Defense Analyses (IDA) Report R-338, Dec 1988

Wolfram, *Mathematica* http://www.wolfram.com/mathematica/

X

XMI: Extensible Markup Language (XML) Metadata Interchange (XMI) http://www.omg.org/spec/XMI/

XML: Extensible Markup Language (XML) http://www.w3.org/TR/REC-xml/

Z

Ziemer, J., Ervin, J., Lang, J., “Exploring Mission Concepts with the JPL Innovation Foundry A-Team,” *AIAA Space 2013 Proceedings*, September 10-12, 2013, San Diego, CA