LIFE-CYCLE MANAGEMENT & DOCUMENTATION CONCEPTS FOR THE SPACE STATION PROGRAM

by

C. P. Barr*, ESL, Inc.
495 Java Drive
Sunnyvale, CA

E. D. Callender, NASA/JPL
NASA HQ, Code SSI
Washington, D.C.

M. J. Steinbacher, NASA/JPL
4800 Oak Grove Drive, Pasadena, CA

ABSTRACT

This paper presents the current life-cycle and documentation concepts under development for the NASA Space Station Program. Specifically, the approaches being taken to address the following concepts are presented: system life-cycle and decomposition concepts; frameworks to establish documentation relationships; and a mechanism to simplify documentation tailoring for specific applications and complexity.

BACKGROUND

Historically, NASA programs have been allocated to a single NASA center with a single prime contractor. Each center has adopted its own standards for software management and acquisition. Due to the increasing size and complexity of the programs and inter-center efforts, NASA has recognized the need for agency-wide standards.

The first step in this process was the establishment of the Software Management and Assurance Program under the direction of the Office of Safety, Reliability, Maintainability, and Quality Assurance (SRM&QA), formerly of the Office of the Chief Engineer. In October 1986, the NASA standard software acquisition life-cycle and a set of documentation standards were baselined. These standards, based on DOD-STD-2167, are specific to software. At a subsequent NASA/Industry conference, industry representatives reviewed these standards and provided three major suggestions:

- o expand the standards to a system perspective
- o define the relationship between the documents
- o define a mechanism for documentation tailoring

The NASA Space Station Program (SSP) has adopted the Office of SRM&QA's software life-cycle and documentation standards for use throughout the program. In addition, they are currently addressing the three suggestions proposed by industry. This paper describes the current status of this effort.

*formerly of Expertware, Inc.

THE SPACE STATION PROGRAM INFORMATION SYSTEMS

The Space Station Program (SSP) is the next major thrust of space exploration by NASA. The initial goal of this program is a permanently manned, low orbit, space station that will be operational by 1994. The space station will provide a platform for experimentation and exploration of space.

The Space Station Program can be viewed as a data or information intensive program consisting of three very large, interrelated information systems: the Technical and Management Information System (TMIS), the Software Support Environment (SSE) System, and the Space Station Information System (SSIS). These support the overall information needs of the Space Station Program.

TMIS is an information system that will provide the management and technical links between the Space Station Program Office in Washington, D.C. and the NASA Centers supporting the Space Station Program (SSP). The SSE System is the SSP software support environment and the necessary communication lines (using TMIS) that are replicated across the program to provide a common software development, integration, and maintenance system.

The central nervous system of the Space Station is the Space Station Information System (SSIS). This system is the integrated, end-to-end family of data handling and processing capabilities supporting all mission and user activities. SSIS users include scientific and applications investigators, flight crew, ground controllers, designers, implementers, and managers of SSP elements. This system contains a permanent core of data handling systems, which are surrounded by application-specific systems that provide the many different user interfaces.

SYSTEM PERSPECTIVE

The complex, software-intensive systems being developed today for Space Station require an approach which addresses the coordination of hardware/network aspects, software aspects, and also operational procedures. In order to accomodate the trade-off analysis and integration process that must be conducted between the hardware, software, and procedure components of a system, an overall system perspective is required. In addition, from a safety and reliability perspective, it is vital that final flight readiness be

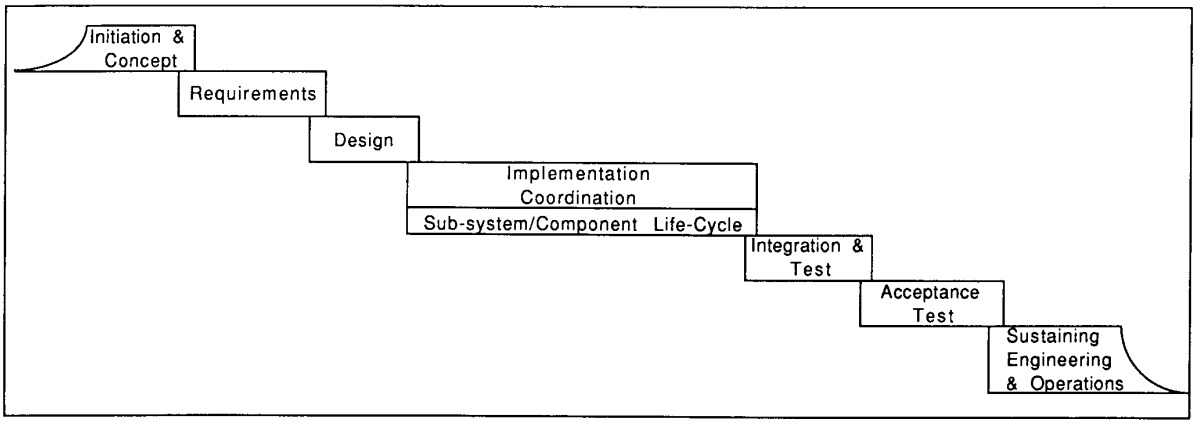

Figure 1. Information System Life Cycle

determined for the system, rather than for its individual components. This information system perspective influences the life-cycle and documentation structure.

Hence, in order to accomodate the system perspective, the life-cycle activities and products are defined for both the system and its components (hardware, software, and operational procedures). The interaction and relationship among the system and the components are defined in terms of a system decomposition methodology. The following sections detail the life-cycle and decomposition aspects of the SSP information standards.

LIFE-CYCLE DEFINITION

A life-cycle is a series of steps, called phases. Within each phase, a set of activities to be performed and the products associated with those activities are defined. The life-cycle that has been defined for SSP information systems is depicted in Figure 1. This life-cycle provides a basic framework, and can be adapted based on development decisions, such as, buy versus build, phased delivery, and incremental development. The adaptation decisions impact the activities to be performed, the products to be developed, and the release/update schedule for the products.

Although the waterfall life-cycle model has its drawbacks, there appears to be no industry-accepted replacement at this time. Figure 2 depicts an example of an adaptation of the basic life-cycle for incremental development. Also depicted is the release time frame for various sections of the technical document. Documentation releases represent snapshots of an activity. The introduction section of a technical document contains the status and schedule information concerning the increments of this development. Figure 3 depicts a similar adaptation for phased delivery. Similar adaptations are required for combined phased delivery/incremental development or other development approaches.

SYSTEM DECOMPOSITION

An information system can be partitioned into lower-level information systems (i.e., subsystems or systems of systems) or into components; this process is called system decomposition. Components are defined as software, hardware, and operational procedures. Components are the lowest level nodes in the decomposition tree, and can be further partitioned into components of the same type. For example, software components may be subdivided to more easily accommodate differing software development approaches or constraints. An individual information system or component represents a node on the system decomposition tree. See Figure 4 for an example of an information system decomposition tree.

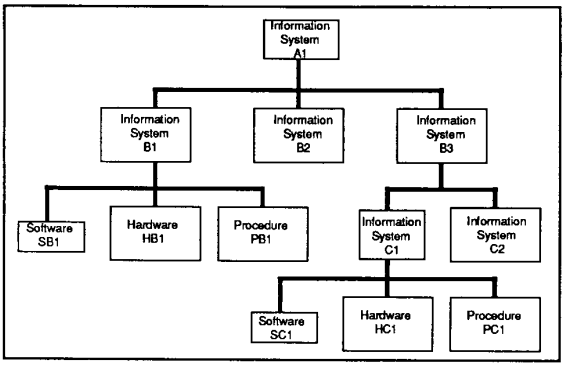

Figure 4. Information System Decomposition Tree

For each node in the system decomposition tree, a life-cycle is instantiated (see Figure 5). Although component life-cycles are based on the information system life-cycle, variations do exist. Figure 6 shows the variations in the life-cycle phases. Several activities, such as certification and sustaining engineering, are, in general, system-level activities.

The life-cycle at each node must be adapted, and the activities and products defined, for that information system or component.

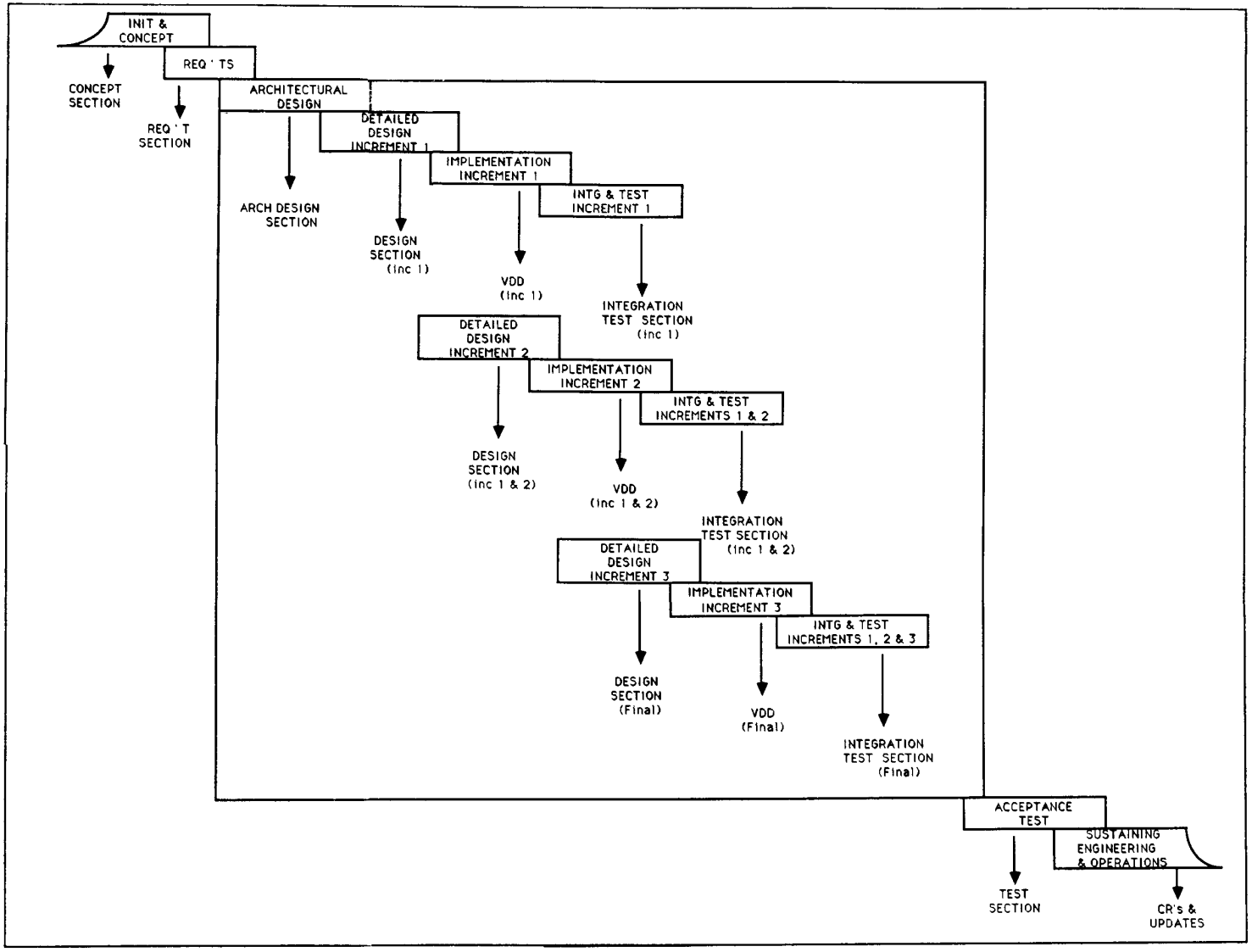

Figure 2. Example of a Software Component Incremental Development Adaptation

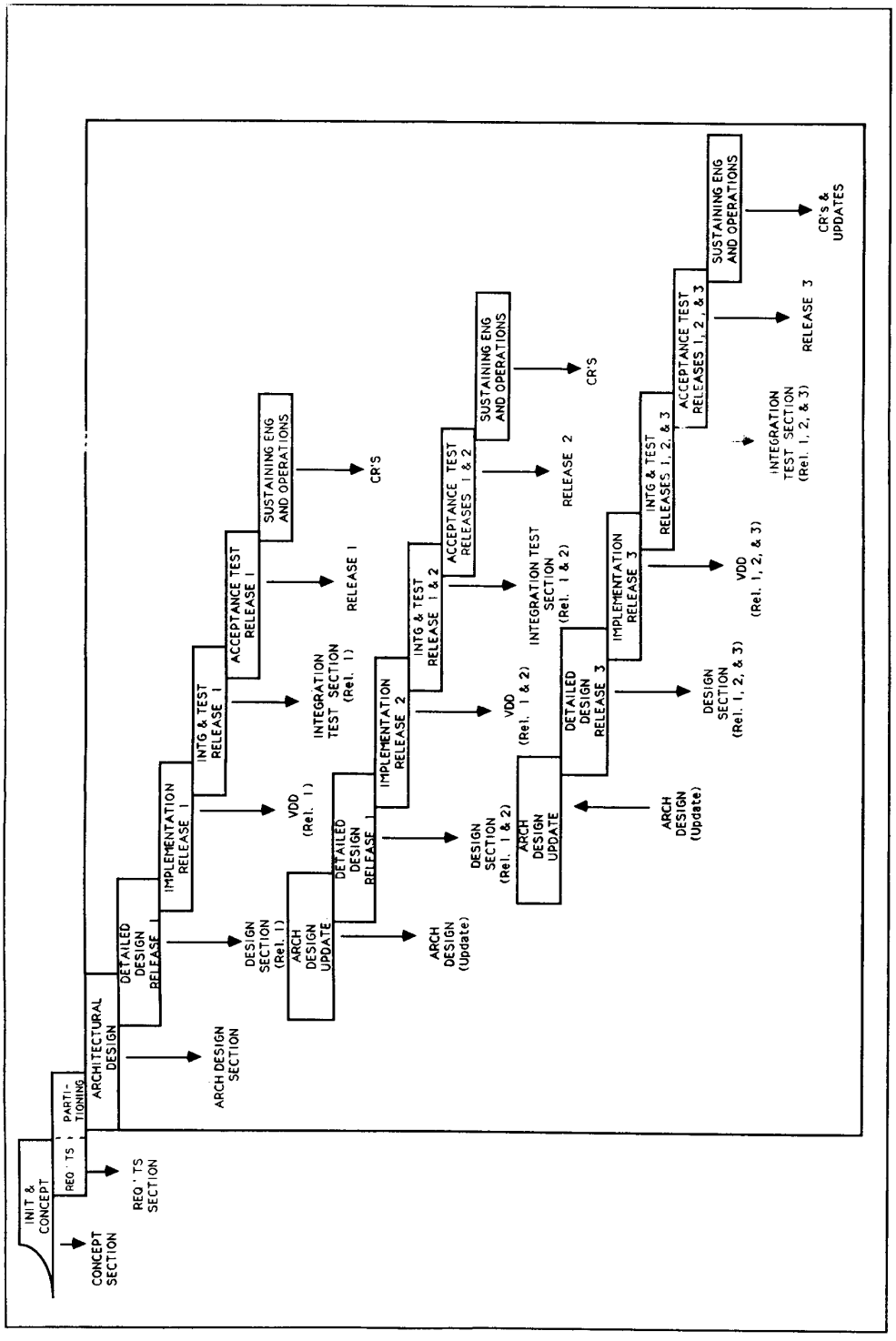

Figure 3. Example of A Software Component Phased Delivery Adaptation

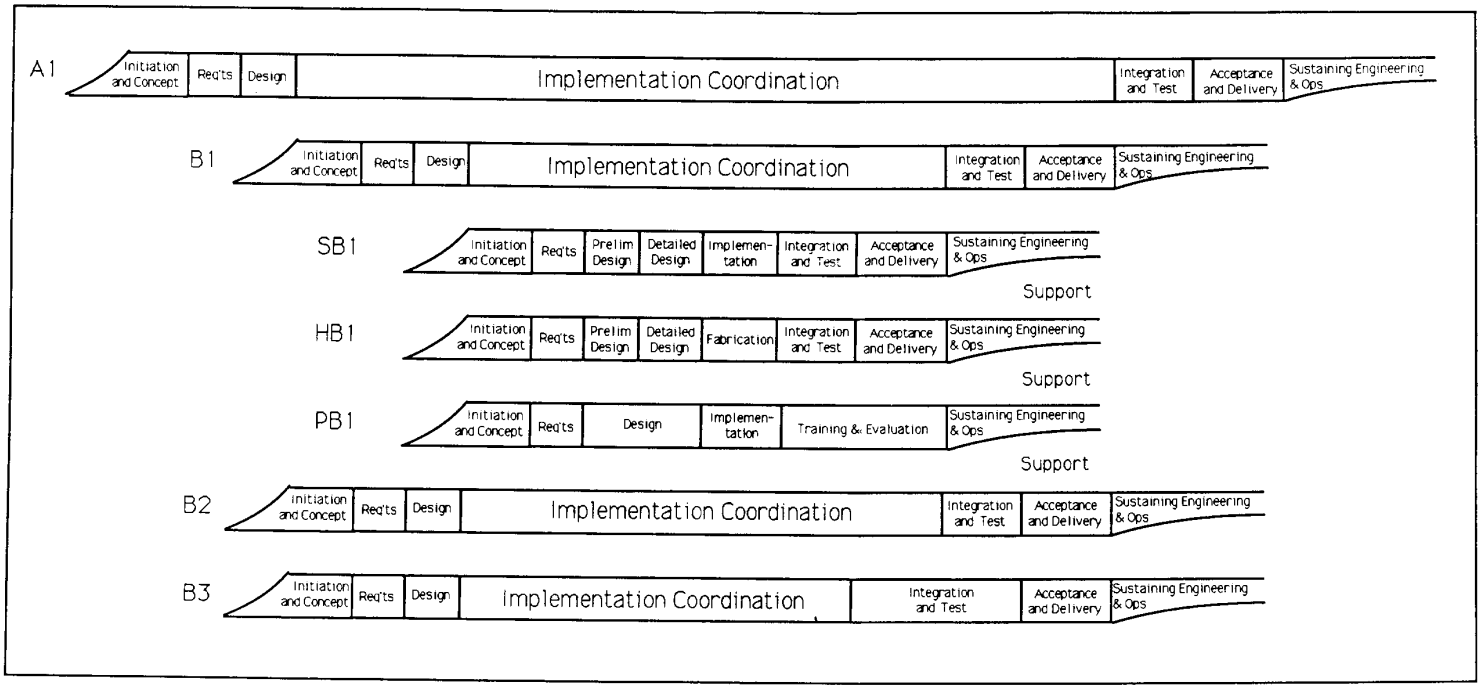

Figure 5. Information System Tree Life-Cycles

However, each information system or component must conform to the standards and requirements specified by its parent(s), i.e., higher-level node(s). For example, incremental development decisions at a node in the system hierarchy may impose phased delivery requirements at lower-level nodes. The introduction section of all documents supports system decomposition traceability by identifying its parent node.

Information System	Software	Hardware	Procedure
Initiation & Concept	Same	Same	Same
Requirements	Same	Same	Same
Design	Prelim Design / Detailed Design	Prelim Design / Detailed Design	Design
Implementation	Same	Fab	Same
Integration & Test	Same	Same	Training & Evaluation
Acceptance & Delivery	Same	Same	
Sustaining Engineering & Operations	(support)	(support)	(support)

Figure 6. Component Life-Cycle Phase Variations

In applications as large and complex as the Space Station Program, the system decomposition, in and of itself, is not a trivial process. A major information system (node) in the SSP decomposition is the Software Support Environment (SSE) System which includes installations at multiple sites, phased deliveries of the software, and extensive interactions required with TMIS. It is vital to further decompose the SSE System into manageable elements. One possible decomposition to the component level of the SSE System is shown in Figure 7. The levels of decomposition and the related life-cycles are highly interdependent. Therefore, the interactions between them must be closely managed. As the complexity of an information system or component increases, so does the need for a hierarchical organization and life-cycle interactions.

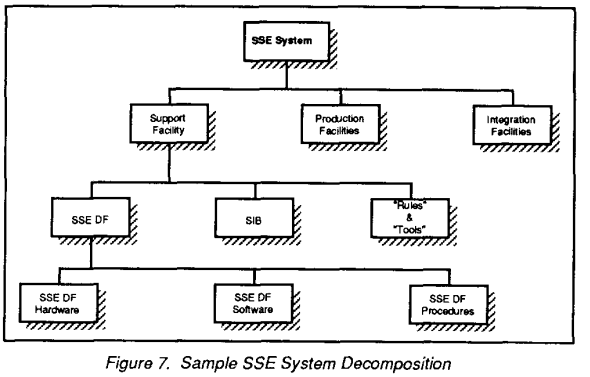

Figure 7. Sample SSE System Decomposition

DOCUMENTATION CONCEPTS

Because of system decomposition levels and the interaction between them, it is important that information concerning each information system or component is well-documented and available to adjacent levels. Due to the number of systems and levels within the Space Station Program, a key consideration is to organize the information to facilitate location of and access to specific data.

The SRM&QA standards, which have been adopted by the Space Station, contain approximately fifty Data Item Descriptions (DIDs) in their documentation set. The majority of these DIDs apply to software documentation only; a few are system oriented. At project start-up time, the magnitude of documentation required is overwhelming, even for the most seasoned manager. In addition, little guidance is currently available for tailoring both the documentation and the life-cycle to the specific project.

In an attempt to reduce the complexity of defining the required documentation for a specific system or component, to provide consistency across the Space Station Program, and to facilitate the access to specific data, a documentation framework was developed. This framework hierarchically organizes all the information contained in the SRM&QA DIDs into a documentation set, thus defining documentation relationships. The documentation set, which applies to each information system and component, includes:

Management Plan
Management Control & Status Reports
Product Specification
Assurance Specification

SSP-specific standards and procedures support this documentation set. The framework approaches documentation by defining the minimum rather than the maximum and, by doing so, supports a document tailoring mechanism called roll-out, described in the following section.

DOCUMENTATION FRAMEWORKS AND TAILORING

The SSP information system standards approach documentation from a minimum set concept. However, it is not always realistic to manage all the information in just four physical documents. Therefore, it is necessary to provide a method to expand the minimum documentation set into multiple volumes. Roll-out simply refers to the mechanism of recording the details of a section in a separate volume, while insuring traceability. The decision to roll-out sections of the documents into separate volumes is the responsibility of the cognizant manager. Often sections are rolled-out into separate volumes if the activities documented in a section are delegated to a separate organization or it helps to manage complexity. When a section is rolled-out, a standard template is used to define the organization of the new volume.

The management plan framework maps the planning and business aspects information included in the NASA SRM&QA DIDs into one hierarchical plan. This framework is depicted in Figure 8. The management plan represents an agreement between

the party obtaining the system and the parties responsible for providing the specific capabilities, such as development. These are referred to as the acquirer and provider(s), respectively. The management plan details the methods and approach for management, acquisition, and assurance for an information system or a component.

"Information system" Management Plan

*1.0 INTRODUCTION
*2.0 APPLICABLE DOCUMENTS
*3.0 RESOURCES, BUDGETS, SCHEDULES & ORGANIZATION
4.0 "information system" ACQUISITION PLAN (37)
5.0 "information system" DEVELOPMENT (03)
5.1 RISK MANAGEMENT (44)
5.2 SECURITY (30)
5.3 ENGINEERING & INTEGRATION (43)
5.4 CONFIGURATION MANAGEMENT (04)
5.4 PRODUCT ASSURANCE (05, 18, 19)
5.6 TRAINING FOR DEVELOPMENT PERSONNEL (36)
5.7 DELIVERY & OPERATIONAL TRANSITION (47)
6.0 "information system" INDEPENDENT VERIFICATION & VALIDATION (32)
7.0 "information system" CERTIFICATION (49)
8.0 "information system" SUSTAINING ENGINEERING & OPERATIONS (42)
9.0 "information system" EVOLUTIONARY ACQUISITION
*10.0 NOTES
*11.0 APPENDICES
*12.0 GLOSSARY

Figure 8. Management Plan Framework

A management plan roll-out example is depicted in Figure 9. A standard template is used to roll-out sections of the management plan. It should be noted that the business aspects (section 3), including work breakdown structure and schedules, are part of this template. This provides a mechanism for recording the allocation of the resources with the associated activity.

The introduction (section 1) contains pointers to the documentation parent document(s) and the rolled-out sections (volumes). In addition, as part of management planning, the roll-out organization of both the product and assurance specifications are defined in the appropriate sections of the management plan. Therefore, the actual number of volumes for any information system or component will depend on management decisions documented in the management plan.

The management control and status reports document records information concerning the feedback and iteration of the development process. The content description is shown in Figure 10.

The product specification documents the technical activities of the development process. It includes the technical concept, requirements, and design of the information system or component. This specification is prepared by the development provider. It records snapshot information of the ongoing engineering process. The framework of the product specification is shown in Figure 11. An example roll-out is depicted in Figure 12.

The assurance specification records all the assurance activities. The content description is shown in Figure 13. The frameworks presented in this section are applicable to information systems. An overview of component framework variations is summarized in Figure 14.

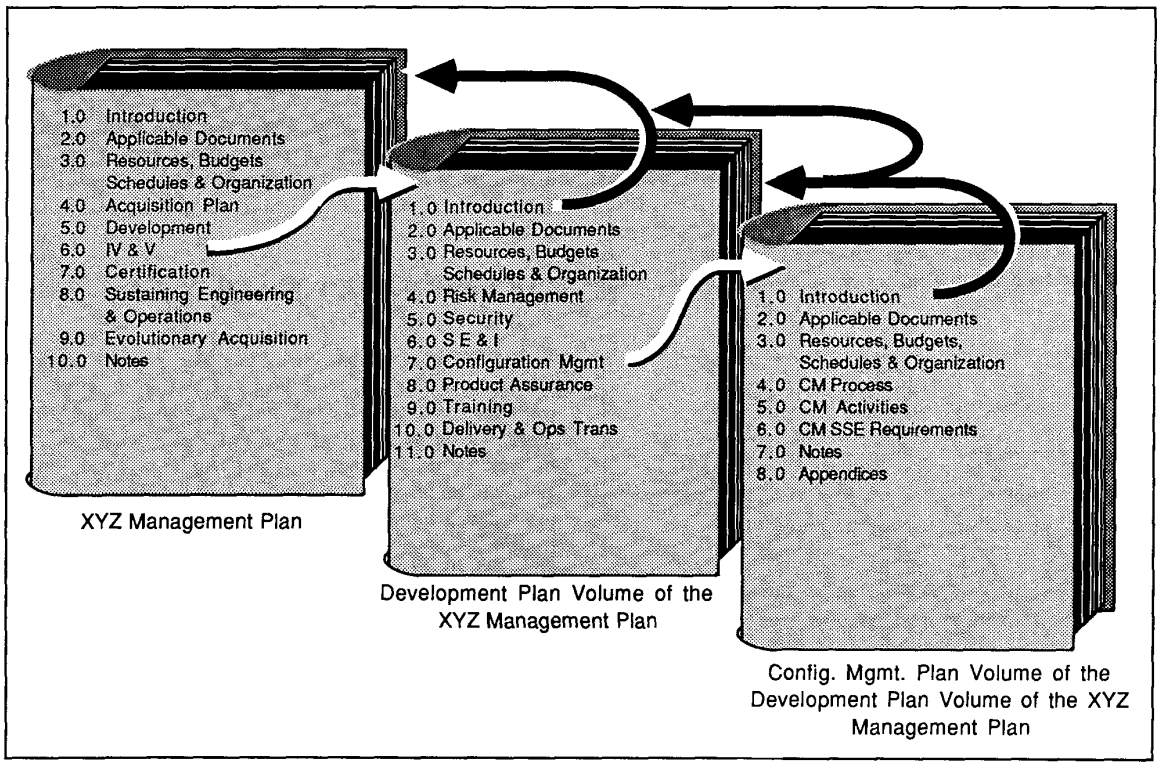

Figure 9. Management Plan Roll-Out Sample

MANAGEMENT CONTROL & STATUS REPORTS CONTENT DESCRIPTION

- STATUS REPORTS
- REVIEW & AUDIT REPORTS (38, 51, 52, 53)
- PERFORMANCE DATA
- WAIVER REQUESTS
- DISCREPANCY/CHANGE REQUESTS (06)
- CHANGE PROPOSALS (29)
- LESSONS-LEARNED (41)

(n) —> Reference to NASA Code Q Version 3.0 DID n

Figure 10. Content Description of the Information System Management Control & Status Reports

"information system" PRODUCT SPECIFICATION

*1.0	INTRODUCTION
*2.0	APPLICABLE DOCUMENTS
3.0	"information system" CONCEPT (01)
4.0	"information system" REQUIREMENTS (00,08)
5.0	"information system" SYSTEM DESIGN (11,13,15)
6.0	"information system" VERSION DESCRIPTION DOCUMENT (16,17)
7.0	"information system" USER'S GUIDE (25)
8.0	"information system" INTEGRATION TEST DOCUMENT
*9.0	NOTES
*10.0	APPENDICES
*11.0	GLOSSARY

*—> Identifies sections that comprise the product specification roll-out template (incomplete; will include, at a minimum, traceability and presentation format/delivery media sections)

(n) —> Reference to NASA Code Q Version 3.0 DID n

"information system" —> Replace with the name of the specific system

Figure 11. Product Specification Framework

CONCLUSIONS AND PLANS

The purpose of this effort was to develop concepts for life-cycle and documentation standards for the NASA Space Station Program. The approach was to utilize the baselined NASA software standards to the extent possible. The objectives included the development of a methodology which would assist managers in the tailoring of the life-cycle for a specific application and in defining the appropriate documentation requirements. Additional objectives included:

- o expanding the standards to a system perspective;
- o defining the relationship between the documents; and
- o defining a mechanism for documentation tailoring.

The concepts presented in this paper represent the approach developed to date. Some testing of this approach through generation of management plans and concept documents has been performed; but more effort in this area is still required. Most of this work was performed prior to the award of the Space Station SSE

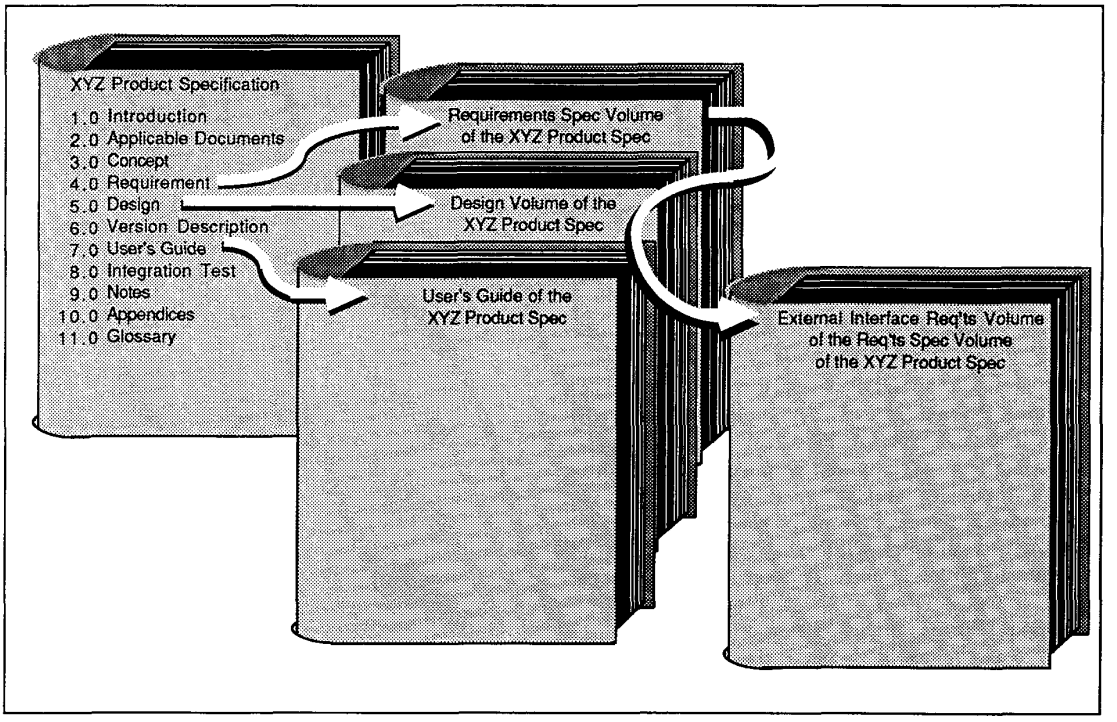

Figure 12. Product Specification Roll-Out Sample

ASSURANCE SPECIFICATION CONTENT DESCRIPTION

- INFORMATION SYSTEM ACCEPTANCE TEST SPECIFICATION, PROCEDURES & RESULTS
  - SPECIFICATION (20)
  - ACCEPTANCE CRITERIA & PROCEDURES (21)
  - RESULTS (23)
- INFORMATION SYSTEM VERIFICATION SPECIFICATION, PROCEDURES & RESULTS
  - SPECIFICATION
  - VERIFICATION CRITERIA & PROCEDURES
  - RESULTS

(Applies to QA, V&V, IV&V, and Certification)

(n) —> Reference to NASA Code Q Version 3.0 DID n

Figure 13. Content Description of the Information System Assurance Specification

Information System	Software	Hardware	Procedure
MANAGEMENT PLAN:	Same	Same	Same
Acquisition Plan	Same	Same	Same
Development	Same	Same	Same
Risk Management	--	--	--
Security	Same	Same	??
Engr & Intg	Same	Same	Same
CM	Same	Same	Same
PA	Same	Same	Same
Training	Same	Same	Same
Delivery & Op trans	Delivery	Delivery	Delivery
IV &V	--	--	--
Certification	--	--	--
Sus engr & Ops	(support)	(support)	(support)
Evolutionary Acq	--	--	--
MANAGEMENT CONTROL & STATUS REPORTS	Same	Same	Same
PRODUCT SPECIFICATION:			
Concept	Same	Same	Same
Requirements	Same	Same	Same
Design	Same	Same	Same
Version Description	Same	Same	Same
User's Guide	Same	Same	Ops Manual
--	Unit Test	Unit Test	--
Integration Test	Same	Same	--
ASSURANCE SPECIFICATION:			
Acceptance Test	Same	Same	Eval Test
Verification Test	--	--	--

Figure 14. Component Life-Cycle Framework Variations

and TMIS contracts. The SSE is the keeper of the "rules and tools" for SSP software development. In order for these standards to be effectively utilized and refined on the SSP, they will need to be incorporated into the support environments, such as the SSE. The framework and roll-out concepts were designed to support the automation of documentation tailoring, preparation, and management. The current plan is for the SSE to utilize these concepts and approach as input into their standards development process for the SSP.

As the Space Station program is a major effort for NASA, lessons learned concerning this standards effort will be incorporated into NASA's agency-wide standards. A high level of interaction between this effort and the Office of SRM&QA has been maintained in order to accommodate this technology transfer.

ACKNOWLEDGMENTS

We would like to acknowledge the efforts of many people who have supported this effort. In particular, the NASA Software Working Group for its early efforts in software policies and standards for the Space Station Program; our own working group, including Barbara Christoph, Mike Evans, Peg Smith, and Sue Voigt for their long hours; and also Jack Garman, Director Information System Services Program Group--Space Station Program Level II, for supporting this development.

REFERENCES

[1] DOD-STD-2167

[2] NASA, Office of SRM&QA, Software Standards, Version 3.0, 15 October 1986

[3] NASA/Industry Software Workshop Proceedings, February 1987