NOT MEASUREMENT SENSITIVE

FAA-HDBK-006A

January 7, 2008

FEDERAL AVIATION ADMINISTRATION HANDBOOK

Reliability, Maintainability, and Availability (RMA) HANDBOOK

This handbook is for guidance only.
Do not cite this document as a requirement.

FOREWORD

1. This is a new handbook. It is approved for use by the Federal Aviation Administration, Department of Transportation. It is also available for use by all other DOT agencies and their personnel.
2. This handbook covers the development of reliability, maintainability and availability (RMA) requirements for the National Airspace System (NAS).
3. This document will guide Service Units and acquisition managers in preparing procurement packages for major system acquisitions. RMA-related sections of these packages include Information for Proposal Preparation, System-Level Specifications, Statements of Work, and Data Item Descriptions. The handbook not only establishes RMA contractual requirements but also recommends comprehensive steps to ensure that fielded systems successfully comply with them. It provides guidance to help managers reduce NAS-Level requirements to levels of detail and characteristics that can readily be monitored and verified. Additionally, it recommends procedures to help managers evaluate proposals, monitor design development, and conduct effective tests and verifications.
4. Comments, suggestions, or questions on this document should be addressed to the Federal Aviation Administration, 800 Independence Ave., S.W., Washington, DC, 20591, System Engineering Office, NAS Requirements and Interface Management Division.

CONTENTS

TABLE OF FIGURES

TABLE PAGE

REVISION HISTORY

Date	Version	Comments
	1.16	Reorganized STLSC matrices to Terminal, En Route, and Other; combined Info and R/D threads for each domain Added power codes to matrices Fixed text and references referring to matrices Added power text.

1 SCOPE

Most of the systems comprising the National Airspace System (NAS) fall into one of three general categories:

- Automated information systems that continuously integrate and update data from remote services to provide timely decision-support services to Air Traffic Control (ATC) specialists
- Remote and distributed subsystems that provide services such as navigation, surveillance, and communications to support NAS ATC systems
- Infrastructure systems that provide services such as power, heating, ventilating, and air conditioning (HVAC) systems, and telecommunications to support NAS facilities

This document primarily allocates NAS-Level requirements to the information systems that provide consolidated ATC services. These systems involve software-intensive air traffic control automation and communications capabilities. They have stringent availability requirements and, as a consequence of the large amounts of custom software that must be developed for them, entail significant cost and schedule risks. These programs provide the most critical operational services and have the most visibility. For these reasons, it is appropriate that they be given the most attention in this handbook.

Remote and distributed subsystems achieve the necessary overall availability through their reliance upon diversity tailored to meet specific regional considerations. The availability of the individual elements comprising these subsystems is furthermore determined by life-cycle considerations, not by top-down allocations from NAS-level requirements.

Because infrastructure systems such as power systems, heating ventilation and air conditioning (HVAC) systems typically violate the independence assumption underlying RMA calculations, they can directly cause failures in the systems they support. Therefore, top-down allocations of availability requirements are not appropriate for these systems. Instead, the aviation community needs to prepare and standardize a new, well defined set of configurations to use with infrastructure systems.

This handbook is for guidance only and cannot be cited as a requirement.

2 APPLICABLE DOCUMENTS

2.1 Government Documents[1]

2.1.1 Specifications, standards, and handbooks

FEDERAL AVIATION ADMINISTRATION

NAS-SS-1000, *FAA System Requirements*, 21 March 1985.

NAS-SR-1000, *FAA System Requirements*, 21 March 1985.

FAA System Engineering Handbook, 19 November 2003.

DEPARTMENT OF DEFENSE

MIL-HDBK-217F, *Reliability Prediction of Electronic Equipment*, 2 December 1991.

MIL-STD-471A, *Maintainability Verification/Demonstration/Evaluation*, 27 March 1973.

MIL-HDBK-472, *Maintainability Prediction*, 24 May 1966.

MIL-STD-498, *Software Development and Documentation*, 5 December 1994.

MIL-STD-721C, *Definition of Terms for Reliability and Maintainability*, 12 June 1981.

MIL-STD-756B, *Reliability Modeling and Prediction*, 18 November 1981.

MIL-STD-781D, *Reliability Testing for Engineering Development, Qualification, and Production,* 18 October 1986.

MIL-STD-882D, *Standard Practice for System Safety*, 10 February 2000.

MIL-STD-1629A, *Military Standard Procedures for Performing a Failure Mode, Effects and Criticality Analysis*, 24 November 1980.

MIL-STD-961E, Department of Defense Standard Practice, *Defense and Program-Unique Specifications Format and Content*.

MIL-STD-967, Department of Defense Standard Practice, *Defense Handbooks Format and Content*, 1 August 2003.

2.1.2 FAA Orders

FAA Order 6040.15C, *National Airspace Reporting System (NAPRS)*, December 23, 1991.

FAA Order 6040.36A, *Communications Diversity*, 11/14/95.

FAA Order 6000.30, *Certification*

[1] Note: Some documents listed in this section may not reflect the most recent version.

FAA Order 6950.2D, *Electrical Power Policy Implementation at National Airspace System Facilities, 10/16/03*.

2.1.3 Other Government documents, drawings, and publications

2.2 Non-Government Publications

En Route Automation Redundancy Study Task, Final Report, March 2000.

Einhorn, S. J., "*Reliability Prediction for Repairable Redundant Systems,*" Proceedings of the IEEE; February, 1963.

3 DEFINITIONS

This section provides definitions of RMA terms used in this document and in the RMA section of the NAS-SR-1000. Three basic categories of definitions are presented in this section:

- Definitions of commonly used RMA terms and effectiveness measures
- Definitions of RMA effectiveness measures tailored to address unique characteristics of FAA fault-tolerant automation systems
- Definitions of unique terms used both in this document and in the RMA section of NAS-SR-1000

Definitions for commonly used RMA effectiveness terms are based on those provided in MIL-STD-721. In some cases, where multiple definitions exist, the standard definitions have been modified or expanded to provide additional clarity or resolve inconsistencies.

For unique terms created during the preparation of the document and the RMA section of the NAS-SR-1000, a brief definition is included along with a pointer to the section of the handbook where the detailed rationale is provided.

This document assumes the reader is familiar with the NAS Architecture (Version 5.0 or greater) and its associated terminology. Readers unfamiliar with the NAS Architecture are referred to the FAA ATO-P System Engineering website: www.faa.gov/asd/.

AVAILABILITY: The probability that a system or constituent piece may be operational during any randomly selected instant of time or, alternatively, the fraction of the total available operating time that the systems or constituent piece is operational. Measured as a probability, availability may be defined in several ways, which allows a variety of issues to be addressed appropriately, including:

Inherent Availability (A_i) – The maximum availability theoretically within the capabilities of the system or constituent piece. Computations of this construct consider only hardware elements and they assume perfect failure coverage, an ideal support environment, and no software or power failures. Scheduled downtime is not included in the Inherent Availability measure. A_I is an inherent design characteristic of a system that is independent of how the system is actually operated and maintained in a real world environment.

Equipment and Service Availability (A_{es}) – Includes all sources of down time associated with unscheduled outages, including logistics and administrative delays, but excludes scheduled downtime. A_{es} is an operational performance measure for deployed systems and is monitored by the National Airspace Reporting System (NAPRS) for all reportable facilities and services.

Operational Availability (A_{op}) – The availability including *all* sources of downtime, both scheduled and unscheduled. A_{op} is an operational measure for deployed systems that is monitored by NAPRS.

CERTIFICATION: A quality control method used by Airways Facilities (AF) to ensure NAS systems and services are performing as expected. AF shall determine certification requirements. AF is authorized to render an independent discretionary judgment about the provision of advertised services. Also because of the need to separate profit motivations from operational decisions and the desire to minimize liability, certification and oversight of the NAS are inherently governmental functions. [FAA Order 6000.30, Definitions Para 11.d]

COVERAGE: Probability of successful recovery from a failure given that a failure occurred.

CRITICALITY: A relative measure of the consequence of a failure mode and its frequency of occurrence.

FACILITY: Generally, any installation of equipment designated to aid in the navigation, communication, or control of air traffic. Specifically, the term denotes the total electronic equipment, power generation, or distribution systems and any structure used to house, support, and /or protect the use equipment and systems. A facility may include a number of systems, subsystems, or equipment.

FAILURE: The event or inoperable state in which any item or part of an item does not, or would not perform as previously specified.

- Dependent Failure: A failure caused by the failure of an associated item(s).
- Independent Failure: A failure that is not caused by the failure of any other item.

FAILURE MODE AND EFFECTS ANALYSIS (FMEA): A procedure for analyzing each potential failure mode in a system to determine its overall results or effects on the system and to classify each potential failure mode according to its severity.

Failure Rate: The total number of failures within an item population, divided by the total number of operating hours.

FAULT AVOIDANCE: The objective of fault avoidance is to produce fault free software. This activity encompasses a variety of techniques that share the objective of reducing the number of latent defects in software programs. These techniques include precise (or formal) specification practices, programming disciplines such as information hiding and encapsulation, extensive reviews and formal analyses during the development process, and rigorous testing.

FAULT TOLERANCE: Fault tolerance is an attribute of a system that is capable of automatically detecting, isolating, and recovering from unexpected hardware or software failures.

INDEPENDENT SERVICE THREAD PAIRS: Because independent Service Thread pairs entail two Service Threads composed of separate system components that provide alternate data paths, they provide levels of reliability and availability that cannot be achieved with a single Service Thread. Such threads may share a single power source. To do so, however, that power source must be designed, or the power system topology must be configured, to minimize failures that could cause both threads to fail. The independent thread pairs may share displays, provided adequate redundant displays are provided to permit the specialist to relocate to an alternate display in the event of a display failure. Independent Service Thread pairs may or may not require diverse hardware and software, but both threads should be active and available at all times. Users need to be able to select either thread at will without need for a system switchover (See Section 6.3 for a detailed discussion of Independent Service Thread pairs.)

INHERENT VALUE: A measure of reliability, maintainability, or availability that includes only the effects of an item's hardware design and its application, and assumes an ideal operation and support environment functioning with perfect software.

LOWEST REPLACEABLE UNIT (LRU): For restoration purposes, an LRU is an assembly, printed circuit board, or chassis-mounted component that can easily be removed and replaced.

MAINTAINABILITY: The measure of the ability of an item to be retained in or restored to specified condition through maintenance performed, at each prescribed level of maintenance and repair, by appropriately skilled personnel using prescribed procedures and resources.

Many maintainability effectiveness measures have inconsistent and conflicting definitions, and the same acronym sometimes represents more than one measure. These inconsistencies generally arise as a consequence of the categories of downtime that are included in a maintainability effectiveness measure. The following definitions reflect the usage in this document and the NAS-SR-1000:

- **Mean Time to Repair (MTTR)** – Mean Time to Repair is a basic measure of maintainability. It is the sum of corrective maintenance times (required at any specific level of repair) divided by the total number of failures experienced by an item that is prepared at that level, during a particular interval, and under stated conditions. The MTTR is an inherent design characteristic of the equipment. Traditionally, this characteristic represents an average of the number of times needed to diagnose, remove, and replace failed hardware components. In effect, it is a measure of the extent to which physical characteristics of the equipment facilitate access to failed components in combination with the effectiveness of diagnostics and built in test equipment.

  MTTR is predicted by inserting a broad range of failed components and measuring the times to diagnose and replace them. It is calculated by statistically combining the component failure rates and the measured repair times for each component. The measure assumes an ideal support environment in which trained technicians with all necessary tools and spare parts are immediately available – but it does not include scheduled downtime for preventive maintenance or such things as the time needed for a technician to arrive on scene or delays in obtaining necessary spare parts. With the increasing use of Commercial Off-the-Shelf (COTS) equipment, MTTR is becoming less significant in System-Level specifications because it is a predetermined, inherent characteristic of the manufacturer's design.

- **Mean Time to Restore Service (MTTRS) –** The MTTRS is also an inherent measure of the design characteristics of complex systems. It represents the time needed to manually restore service following an unscheduled service failure requiring manual intervention. Like MTTR, it includes only unscheduled downtime and assumes an ideal support environment, but the MTTRS includes not only the time for hardware replacements, but also times for software reloading and system restart times. MTTRS does not include the times for the successful operation of automatic fault detection and recovery mechanisms that may be part of the system design. The performance specifications for the operation of automatic recovery mechanisms are addressed separately.

- **Mean Down Time (MDT) –** Mean Down Time is an operational performance measure that includes all sources of system downtime, including corrective maintenance, preventive maintenance, travel time, administrative delays, and logistics supply time.

- **MAINTENANCE SIGNIFICANT ITEMS (MSI)** – Hardware elements that are difficult to replace, i.e., cables, backplanes, and antennas.

- **MEAN TIME BETWEEN OUTAGE (MTBO) –** MTBO is an operational performance measure for deployed systems that corresponds to the inherent MTBF measure. A measure of the time between unscheduled interruptions, MTBO is monitored by NAPRS. It is computed by dividing the total operating hours by the number of outages.

NAS CAPABILITY CRITICALITY: each of the NAS Architecture Capabilities is assigned one of three criticality ratings with an associated inherent availability requirement. The NAS capability criticalities are:

CRITICAL (.99999) Loss of this capability would raise to an unacceptable level, the risk associated with providing safe and efficient local NAS operations.

ESSENTIAL (.999) Loss of this capability would significantly raise the risk associated with providing safe and efficient local NAS operations.

ROUTINE (.99) loss of this capability would have a minor impact on the risk associated with providing safe and efficient local NAS operations.

NON-DEVELOPMENTAL ITEM (NDI): An NDI is system, or element of a system, that is used in a developmental program but has been developed under a previous program or by a commercial enterprise.

RECOVERY TIME: For systems that employ redundancy and automatic recovery, the total time required to detect, isolate, and recover from failures. Recovery time is a performance requirement. While successful automatic recoveries occurring within the prescribed recovery time are not counted as downtime in RMA computations, requirements for systems employing automatic recovery do limit the allowable frequency of automatic recovery actions.

RELIABILITY: Reliability can be expressed either as the probability that an item or system will operate in a satisfactory manner for a specified period of time, or, when used under stated conditions, in terms of its Mean Time between Failures (MTBF). Expressing reliability as a probability is more appropriate for systems such as missile systems that have a finite mission time. For repairable systems that must operate continuously, reliability is usually expressed as the probability that a system will perform a required function under specific conditions for a stated period of time. It is a function of (MTBF), according to the formula

$$R = e^{-\frac{t}{m}}$$

where “t” is the mission time and “m” is the MTBF. Also, reliability is often expressed as the raw MTBF value, in hours, rather than calculating R according to the above formula.

SYSTEM ANALYSIS AND RECORDING (SAR): A system function that records significant system events, performance data, and system resource utilization for the off-line analysis and evaluation of system performance. Typical data to be recorded includes:

a. All system inputs

b. All system outputs

c. All system and component recoveries and reconfigurations

d. System status and configuration data including changes

e. Performance and resource utilization of the system and system components

f. Significant security events

SERVICE: The term “service” has different meanings in the contexts of the NAS Architecture (Version 5.0 or greater) and the NAPRS.

NAS Architecture Service represents services, such as separation assurance, that are provided to NAS users. These services are provided by a combination of ATC specialists and the systems that support them. Each NAS Architecture Service comprises two or more NAS capabilities associated with the service.

> **NAPRS Services** as defined in FAA order 6040.15 are services that represent an end product, which is delivered to a user (AT personnel, the aviation public, or military) that results from an appropriate combination of systems, subsystems, equipment, and facilities.

To distinguish the NAPRS services from NAS Architecture Services, NAPRS services will be referred to in this document as "Service Threads."

SERVICE THREADS: Service Threads are strings of systems that support one or more of the NAS Architecture Capabilities. These Service Threads represent specific data paths (e.g. radar surveillance data) to controllers or pilots. The threads are defined in terms of narratives and Reliability Block Diagrams depicting the systems that comprise them. They are based on the reportable services defined in FAA Order 6040.15D National Airspace Performance Reporting System (NAPRS). Note that some new Service Threads have been added to the set of NAPRS services, and some of the NAPRS services that are components of higher-level threads have been removed. (See section 6.2 for a detailed discussion of the Service Thread concept.)

SERVICE THREAD LOSS SEVERITY CATEGORY (STLSC): Each Service Thread is assigned one of three Service Thread Loss Severity Categories based on the severity of impact that loss of the thread could have on the safe and efficient operation and control of aircraft. (See Section 6.4 for a detailed discussion of the STLSC concept.) The Service Thread Loss Severity Categories are:

> Safety-Critical –Service thread loss would present an unacceptable safety hazard during transition to reduced capacity operations.
>
> Efficiency-Critical – Service thread loss could be accommodated by reducing capacity without compromising safety, but the resulting impact has the potential for system-wide impact on NAS efficiency of operations.
>
> Essential – Service thread loss could be accommodated by reducing capacity without compromising safety, with only localized impact on NAS efficiency.

SYSTEM STATUS INDICATIONS (e.g., ALARM, RETURN-TO-NORMAL): Indications in the form of display messages, physical or graphical indicators, and/or aural alerts designed to communicate a change of status of one or more system elements.

TARGET OPERATIONAL AVAILABILITY: The desired operational availability associated with a given NAS Service/Capability Criticality.

VALIDATION: The process of applying a methodology to determine that the right system is being built (i.e., that the system requirements are unambiguous, correct, complete, consistent, operationally and technically feasible, and verifiable).

VERIFICATION: The process of applying a methodology to determine that the design solution has met the system requirements and that the system is ready for use in the operational environment for which it is intended.

4 GENERAL GUIDANCE

This handbook is intended to assist FAA Service Units and acquisition managers in the preparation of the RMA sections of procurement packages for major system acquisitions. These sections include System-Level Specifications (SLS), Statements of Work (SOWs), Information for Proposal Preparation (IFPP) documents, and associated Data Item Descriptions (DIDs). The document provides guidance for the decomposition of NAS-Level requirements to produce detailed specifications and characteristics that can be readily monitored and verified. Recommended procedures for evaluation of contractor proposals, monitoring of the design development, and testing and verification are included. The intent is to present a comprehensive set of steps that not only establish contractual requirements, but also helps to ensure the achievement of operationally acceptable reliability, maintainability, and availability characteristics in the fielded systems.

The document also provides guidance to headquarters system engineering personnel responsible for maintaining the NAS SR-1000 requirements related to Service Threads.

4.1 Purpose and Objectives

The NAS-Level requirements and this handbook are based on the NAS Architecture. This handbook uses the NAS Architecture terminology (Version 5.0 or greater) throughout and differentiates “overloaded” terms that have one definition in the context of the NAS Architecture and a different definition elsewhere in the FAA.

4.1.1 Purpose of NAS-Level RMA Requirements

The primary purpose of defining NAS-Level RMA requirements is to relate NAS Architecture Capability functional requirements to verifiable specifications for the hardware and software systems that support these capabilities. An intermediate step in this process is the introduction of the concept of generic Service Threads that define specific services, provided to controllers and/or pilots, which support the various NAS Architecture Capabilities. The Service Threads serve to bridge the gap between un-allocated functional requirements and the specifications for the systems that support them. They also provide the vehicle for allocating NAS-Level RMA-related[2] requirements to specifications for the systems that comprise the Service Threads.

NAS-Level RMA requirements are provided to satisfy the following objectives:

- Provide a bridge between NAS-Level user needs and System-Level Specifications.
- Establish a common framework upon which to justify future additions and deletions of requirements.
- Provide uniformity and consistency of requirements across procured systems, promoting common understanding among the specifying engineers and the development contractors.
- Establish and maintain a baseline for validation and improvement of the RMA characteristics of fielded systems.

[2] The term “RMA-related requirement(s)” includes, in addition to the standard reliability, maintainability and availability requirements, other design characteristics that contribute to the overall system reliability and availability in a more general sense (e.g., fail-over time for redundant systems, frequency of execution of fault isolation and detection mechanisms, system monitoring and control. on line diagnostics, etc.).

4.1.2 Purpose of this Handbook

This handbook provides comprehensive guidance on how to:

- Interpret and allocate the NAS-SR-1000 NAS-Level RMA requirements to systems.
- Decompose the NAS-Level RMA requirements into realistic and achievable System-Level specifications and design characteristics.
- Establish risk management activities to permit the monitoring of critical fault tolerance and RMA characteristics during system design and development.
- Establish a reliability growth program to ensure that latent design defects are systematically exposed and corrected during testing at the contractor's plant, FAA Technical Center and subsequent deployment.
- Describe a process of updating and maintaining the NAS-Level RMA requirements definition process.

This is intended to be a living document. It will be updated periodically to reflect changes to NAS requirements as well as to incorporate the experience gained from using techniques described in it and from downstream procurements and implementations.

4.2 Document Organization

This handbook covers two major topics. The first, contained in Sections 5 and 6, describes the process used by the NAS Requirements and Interface Management Division to derive the NAS-Level RMA requirements. Section 5 describes the motivation and rationale for departing from the traditional methods for reliability modeling, allocation, prediction and verification techniques developed in the 1960's and provides a summary of the characteristics of the approach described here. Section 6 describes how the NAS-Level RMA requirements were developed. This material is included to provide the background information necessary to develop an understanding of the requirements.

The second major topic, contained in Section 7, addresses the specific tasks to be performed by Service Units, acquisition managers, and their technical support personnel to apply the NAS-Level requirements to major system acquisitions. The section is organized in the order of a typical procurement action. It provides a detailed discussion of specific RMA activities associated with the development of a procurement package and continues throughout the acquisition cycle until the system has successfully been deployed. The approach is designed to help to ensure that the specified RMA characteristics are actually realized in fielded systems.

The elements of this approach are summarized below:

Section 5: A NEW APPROACH – Describes the traditional approach to RMA specification and verification that has been employed since the 1960's, details the issues associated with applying these methods to high reliability, software-intensive systems, and outlines the new approach to RMA specification and verification.

Section 6: DERIVATION OF NAS-LEVEL RMA REQUIREMENTS – Introduces the concept of a Service Thread. Documents the procedures used to map NAS Architecture functional requirements to generic Service Threads to serve as the basis for allocating the requirements to specific systems.

Section 7: ACQUISITION STRATEGIES AND GUIDANCE – Describes the specific tasks to be performed by technical staffs of FAA Service Units and acquisition managers to apply the NAS-Level

requirements to System-Level Specifications and provides guidance and examples for the preparation of RMA portions of the procurement package.

- 7.1: Preliminary Requirements Analysis
- 7.2: Procurement Package Preparation
  - i. System-Level Specification (SLS)
  - ii. Statement of Work (SOW)
  - iii. Information for Proposal Preparation (IFPP)
- 7.3: Proposal Evaluation
  - i. Reliability Modeling and Assessment
  - ii. Fault-Tolerant Design Evaluation
- 7.4: Contractor Design Monitoring
  - i. Formal Design Reviews
  - ii. Technical Interchange Meetings
  - iii. Risk Management
- 7.5: Design Validation and Acceptance Testing
  - i. Fault Tolerance Diagnostic Testing
  - ii. Functional Testing

Section 8: NAS-SR-1000 MAINTENANCE – Describes the process for updating the Service Thread database to maintain consistency with the NAPRS services in response to the introduction of new services, system deployments, modifications to NAPRS services, etc.

- i. Revising a Service Thread's RMA requirements
- ii. Adding a new Service Thread

Section 9: RMA REQUIREMENTS ASSESSMENT – Describes the approach used to compare new requirements with the performance of fielded systems to verify the achievability of proposed requirements, ensure that the reliability of new systems will be at least as good as that of existing systems, and to identify deficiencies in the performance of currently fielded systems.

5 A NEW APPROACH

The tools and techniques that are the foundation for reliability management were developed in the late 1950's and early 1960's. In that timeframe, the pressures of the cold war and the space race led to increasing complexity of electronic equipment, which in turn created reliability problems that were exacerbated by the applications of these "equipments" in missile and space applications that did not permit repair of failed hardware. This section will examine the traditional approaches to RMA specification and verification and describe how changes that have occurred over the past four decades have created a need for a dramatic change in the way these RMA issues are addressed.

5.1 The Traditional RMA Paradigm

The FAA has traditionally viewed RMA requirements in a legalistic sense. The requirements have been part of binding contracts with which contractors have been legally obligated to comply.

Because *actual* RMA performance could only be determined after a system was installed, a contractor's *prospective* ability to comply with RMA requirements was evaluated using the predictions of models. Reliability predictions were based on the numbers of discrete components used in these systems and their associated failure rates. A catalog of failure rates for standard components was published in MIL-HDBK-217. These failure rates were based on hundreds of thousands of hours of operating time.

The predicted reliability of equipment still under development was estimated by extrapolating from attested failure rates with adjustments reflecting the numbers and types of components used in the new piece of equipment. If the predicted reliability was unacceptable, engineers used various screening techniques to try to reduce the failure rates of the components. These compensatory efforts generally increased the costs of equipment built to military specifications, and despite efforts to improve reliability, complex electronic equipment often had MTBFs of fewer than 1,000 hours.

To verify that electronic equipment was compliant with the specified reliability, several preproduction models were often placed in a sealed room for a period of time. There, statistical decision methods, as described in MIL-STD-781, were employed to decide whether the requirements actually were met and the design was suitable for release to full production.

Maintainability requirements were verified by statistical techniques such as those defined in MIL-HDBK-472. These techniques involved statistically combining component failure rates with actually measured times to identify, remove, and replace a sample of inserted failed components.

The military standards and handbooks that defined the statistical methods used for predicting and verifying reliability and maintainability were based on well-established concepts that could be found in any introductory textbook on engineering statistics.

5.2 Agents of Change

Several factors have tended to make the traditional paradigm obsolete. Among these are:

- Dramatic increases in system reliability resulting both from a combination of technology advances, the use of redundancy, and application of fault tolerance techniques.
- Fundamental statistical limitations associated with reliability prediction and verification for high reliability systems.

- Difficulties associated with the use of availability as a contractual specification.
- Increased use of software intensive digital systems.
- Emphasis on use Commercial Off-the-Shelf (COTS) hardware.

The implications of these changes on traditional RMA practices and policies are discussed below.

5.2.1 Technology and Requirements Driven Reliability Improvements

Since the 1960's, advances in microelectronics and large scale integration have increased the reliability of digital hardware by almost two orders of magnitude. When the FAA first began to acquire digital systems in the 1960's, the hardware elements typically had reliabilities around 1000 hours. Over the years, technology advancements in integrated circuits have yielded dramatic improvements in the reliability of digital hardware. Greater use of automation in critical applications increased the reliability requirements for these systems, and the increased requirements exceeded the improvements resulting from microelectronics technology advances alone. Redundancy and fault tolerance techniques were employed to further increase system reliability. FIGURE 5-1 summarizes NAPRS data for FY1999 through FY 2004 that illustrates the dramatic improvement of system reliability in the past 40 years.

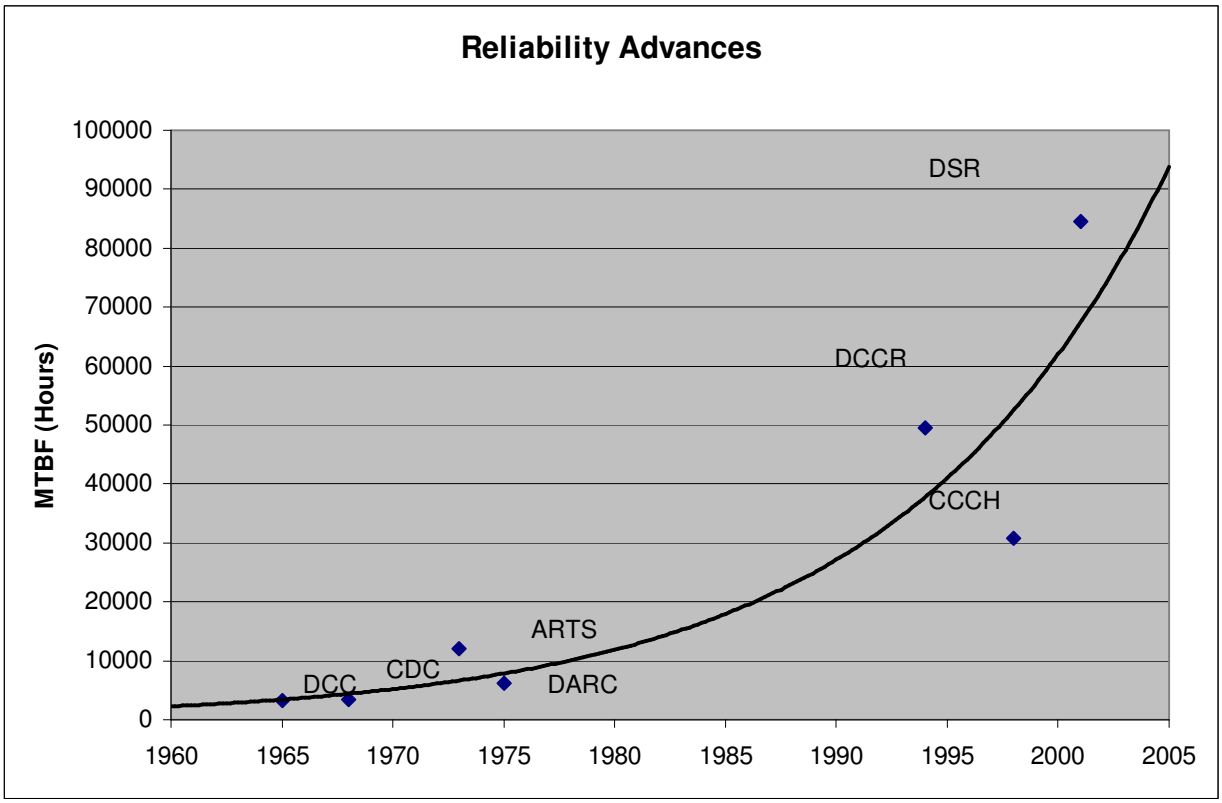

FIGURE 5-1: FAA System Reliability Improvements

5.2.2 Fundamental Statistical Limitations

Since statistical methods were first applied to RMA modeling, allocation, prediction, and verification, there has been an exponential growth in the reliability of digital hardware. There has also been with a related growth in demand for higher reliability in systems for use in critical ATC applications. This exponential growth in the reliability of FAA systems has certainly benefited their users, but it also has created significant challenges to those who specify, predict, and verify the RMA characteristics of these systems. Conventional statistical methods, those that have traditionally been used for these purposes, simply do not scale well to high levels of reliability.

5.2.2.1 Reliability Modeling

Forty years ago, the use of digital computers to process surveillance data and other important real-time and near-real-time operations created a demand for more reliable systems. When the FAA began to acquire NAS En Route Stage A in the early 1960's, the IBM 360 series computer elements that were used in the Central Computer Complex had an MTBF on the order of 1000 hours, and the combined MTBF of all of the elements in the Central Computer Complex (CCC) was predicted to be approximately 60 hours. In contrast, the required reliability for the NAS Stage A CCC was 10,000 hours. The FAA and IBM had to try to achieve this unheard of level of reliability with hardware elements whose reliability was an order of magnitude less than the requirement. They found a way. Together, the Agency and the researchers pioneered the use of redundancy and automatic fault detection and recovery techniques.

The reliability of the CCC was predicted using a set of Markov-based combinatorial equations developed by S. J. Einhorn in 1963. Drawing on the MTBF and MTTR together with the amount of redundancy, the equations predicted the reliability of repairable redundant configurations of identical elements. They modeled the average time taken from the depletion of spares, with resulting failures, and the return to service of failed elements Einhorn's equations were based solely on the combinatorial probably of running out of spare elements and assumed perfect fault coverage, perfect software, and perfect switchover. They did not address the effectiveness of the automatic fault detection and recovery mechanisms or the effect of software failures on the predicted MTBF.

A simple sensitivity analysis of the effectiveness parameter for automatic fault detection and recovery mechanisms yielded a graph such as the one shown in FIGURE 5-2. At 100% effectiveness the CCC 10000 hour reliability requirement is exceeded by 50%. At 0% effectiveness, the predicted CCC reliability would be 60 hours. The true MTBF lies somewhere in between, but the reliability falls so quickly when the fault detection and recovery effectiveness is less than perfect, that it is virtually impossible to predict it with enough accuracy to be useful. Developers include fault handling provisions for all known failure classes in an attempt to achieve 100% effectiveness, but they know that without access to the number of unknown failure modes they are unlikely to reach this goal.[3]

The models used to predict reliability for fault-tolerant systems are so sensitive to the effectiveness parameter for the fault tolerance mechanisms that their predictions have little credibility, and thus little value in forecasting the real-world reliability characteristics of these systems.

[3]Note: as reliability models became more sophisticated and computerized, this parameter became known as "coverage," the probability that recovery from a failure will be successful, given that a failure has occurred. The concepts and underlying mathematics for reliability and availability models is discussed in greater detail in Appendix C

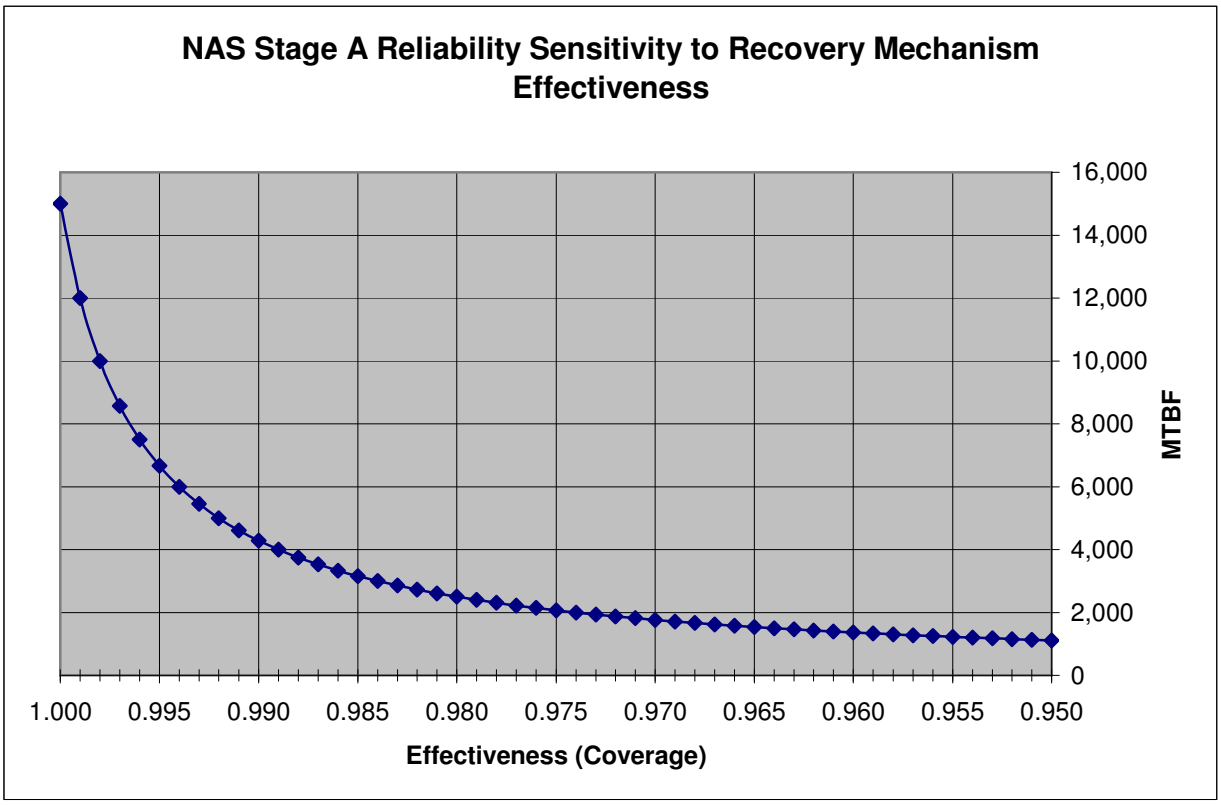

FIGURE 5-2: NAS Stage A Recovery Effectiveness

FIGURE 5-2 shows that although the inherent availability of the redundant configuration exceeds the 10,000 hour requirement by 50%, if the effectiveness of the recovery mechanisms falls below 99.8%, the predicted reliability will not meet the requirement. For this reason, the model cannot be used to predict compliance with a 10,000 hour MTBF requirement.

For a more modern example, consider a hypothetical redundant configuration of two computer servers with MTBFs of 30,000 hours each and a MTTR of 0.5 hours. Although this configuration has a theoretical inherent reliability of one billion hours, the chart in FIGURE 5-3 shows that when coverage drops from 100% to 99%, the predicted reliability drops from one billion hours to 1.5 million hours. At a coverage level of 95%, the predicted reliability drops to 300,000 hours. (Note that the MTBF axis of the chart is logarithmic.)

Although a 300,000 hour MTBF with a fault coverage of 95% should be more than adequate for FAA requirements, there is no assurance that this level will be achieved. If the assumed coverage level is reduced to a more conservative value of 85%, the predicted reliability is still 100,000 hours. This analysis underscores the fact that constructing elaborate and complex mathematical computer models is unnecessary when it can be shown that the model results are almost entirely dependent on an input parameter whose value is essentially either a guess or a value the model has itself derived precisely to get the desired result. The inability to estimate coverage accurately makes it virtually impossible, when using automatic fault detection and recovery mechanisms, to predict the reliability of redundant configurations with enough accuracy to be useful.

Another important conclusion that can be drawn from FIGURE 5-3 is that simple combinatorial models are adequate to verify the theoretical inherent reliability capabilities of the hardware architecture to meet

the requirements. Predicting inherent reliability and availability should be viewed as simply the first step, among many, in evaluating a contractor's ability to meet the RMA requirements.

The conclusion is evident: there is no significant additional benefit to be gained from spending program resources on developing sophisticated computer models. Scarce resources can be better applied toward developing and applying tools and techniques to find and remove latent defects in the recovery mechanisms and software applications that could keep the system from achieving its theoretical maximum. Tools developed under the program should be delivered to the FAA for their use after system acceptance.

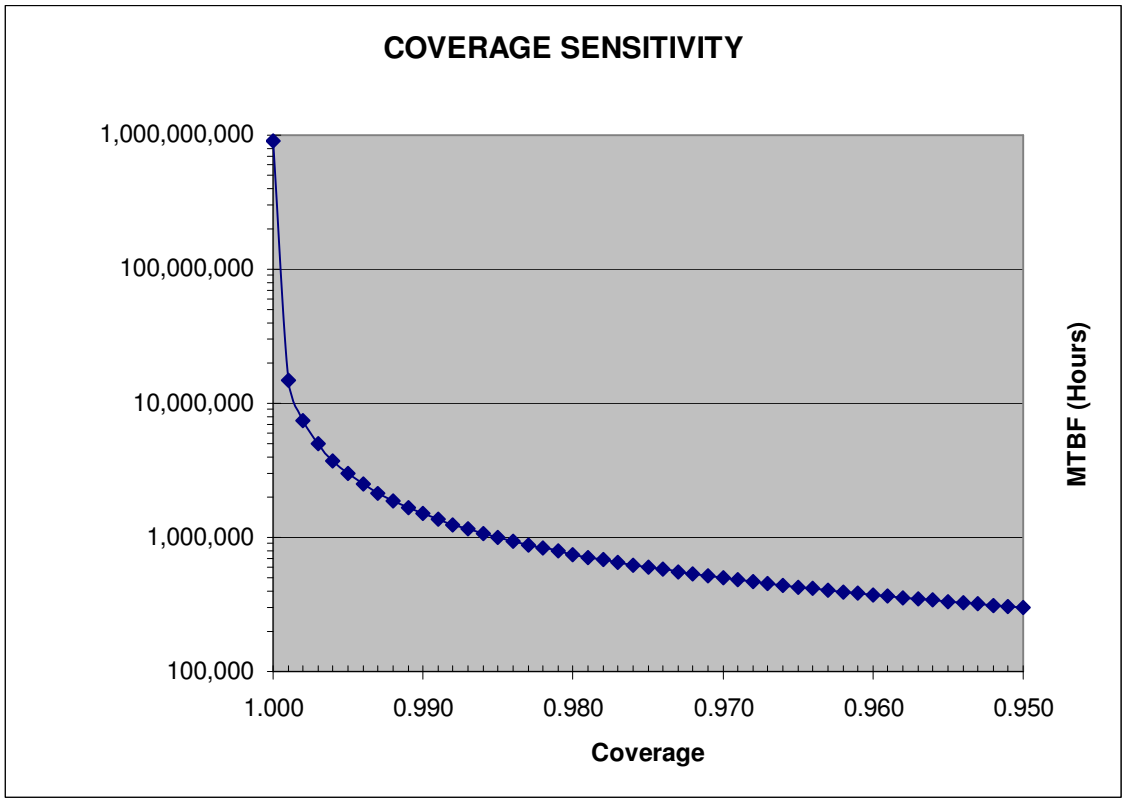

FIGURE 5-3: Coverage Sensitivity of Reliability Models

5.2.2.2 Reliability Verification and Demonstration

The preceding section illustrated some difficulties in predicting the reliability and availability of proposed systems before they are developed. Fundamental statistical limitations also make it difficult to verify the reliability or availability of requirements-driven systems *after* they are developed. Although statistical applications work best with large sample sizes, reliability testing generally obtains limited samples of failures over limited test intervals. High reliability systems seldom fail; therefore, it is impractical to accumulate enough operating hours to obtain a statistically valid sample of failures. A "rule of thumb" is that to obtain a statistically valid sample, the number of test hours should be approximately ten times the required MTBF. For example, a 30,000 hour MTBF system should test either a single system for over 30 years, or test 30 systems for one year. Neither of these alternatives is realistic in the context of a major system acquisition.

Several quantitative parameters are used to describe the characteristics of a formal reliability qualification test, including confidence intervals, producer's risk, consumer's risk, and discrimination ratio. The end result, however, is that – when an accept/reject decision is based on inadequate test time – there is a significant probability of either accepting a system that does not meet the requirements (consumer's risk), or of rejecting a system that does, in fact, meet the requirements (producer's risk).[4]

Arguments underlying these decisions are based strictly on conventional text-book statistics theory. They fail to address the practical reality that modern software systems are not suited to evaluation by fixed reliability qualification tests alone. Today's software is dynamic and adaptive. Enhancements, program trouble reports, patches, and the like present an ever-changing reality that must be effectively managed. The only practical alternative, in today's world, is to pursue an aggressive reliability growth program and deploy a system to the field only when it can be shown to be more stable than the system it will replace. Formal reliability demonstration programs, such as those used for the electronic "black boxes" of the past, are no longer feasible for modern automation systems.

5.2.3 Use of Availability as a Contractual Specification

For the last twenty years, FAA specifications have focused primarily on availability requirements in place of the more traditional reliability and maintainability requirements that preceded them.

Availability requirements are useful at the highest levels of management. They provide a quantitative and consistent way of summarizing the need for continuity of NAS services. They can facilitate the comparison and assessment of architectural alternatives by FAA headquarters system engineering personnel. They also bring a useful performance metric to analyses of operationally deployed systems and Life Cycle Cost tradeoffs. And because it includes all sources of downtime and reflects the perspective of system users, availability is a good overall operational performance measure of the performance of fielded systems.

There are, however, important problems with employing availability as a primary RMA requirement in contractual specifications. This operational performance measure combines equipment reliability and maintainability characteristics with operation and maintenance factors that are beyond the control of the contractor as well as outside of the temporal scope of the contract.

The fundamental concept of availability implies that reliability and maintainability can be traded off. In other words, a one-hour interruption of a critical service that occurs annually is seen as equivalent to a 15-second interruption of the same service that occurs every couple of days, for both scenarios provide approximately the same availability. It should be obvious that interruptions lasting a few seconds are unlikely to have a major impact on ATC operations, while interruptions lasting an hour or more have the potential to significantly impact traffic flow and safety of operations. Contractors should not be permitted, however, to trade off reliability and maintainability arbitrarily to achieve a specific availability goal. Such tradeoffs have the potential to impact NAS operations adversely. They also allow a readily measured parameter such as recovery time to be traded off against an unrealistic and immeasurable reliability requirement following a logic such as: "It may take two hours to recover from a failure, but it will be 20,000,000 hours between failures, so the availability is still acceptable, i.e., seven 'nines.'

During system development, availability can only be predicted using highly artificial models. Following development, system availability is not easily measured during testing at the William J. Hughes Technical

[4] The basic mathematics underlying these effects is summarized in Appendix D. (For a more detailed discussion of reliability qualification testing, see MIL-STD-781.)

Center (WJHTC). The fundamental statistical sampling limitations associated with high levels of reliability and availability are not the only problem. Availability cannot be measured directly; it can only be calculated from measurements of system downtime and the total operating time.

Deciding how much of the downtime associated with a failure should be included or excluded from calculations of availability could prove to be difficult and contentious. In an operational environment, matters are clear cut: simply dividing the time that a system is operational by the total calendar time yields the availability. In a test environment, however, adjustments are required for downtimes caused by things like administrative delays, lack of spares, and the like – factors that the contractor can not control. A failure review board will be faced with the highly subjective process of deciding which failures are relevant and how much of the associated downtime to include.

For these reasons, the FAA needs to establish specifications that can be more readily monitored during development and measured at the contractor's plant and the WJHTC prior to acceptance of the system.

5.2.4 RMA Issues for Software-Intensive Systems

The contribution of hardware failures to the overall system reliability for a software-intensive system is generally negligible. Software reliability is by far the dominant factor in the overall reliability of these systems. Most models that predict software reliability rely on historical data on the numbers of latent defects per thousand source lines of code (at various stages of development) to discover and recommend the removal of latent software defects. These models are useful for estimating test time, manpower and costs to reduce the fault density to acceptable levels; but they provide no insight into the run-time behavior of the software or the predicted operational reliability of the system.

Although some academic papers have attempted to develop models that can relate fault density to the run-time behavior of software, the accuracy and usefulness of these models is questionable and unproven. Again, the fundamental problem in predicting software reliability is the need to predict, with some degree of certainty, how frequently each latent fault in the code is likely to result in an operational failure. Essentially, this is a function of how often a particular section of code is executed. For routines such as surveillance processing that are scheduled at regular intervals, this perhaps could be feasible. Other areas of code, however, may only be executed rarely. For a complex system containing a million or more lines of code, with various frequencies of occurrence, the prediction problem becomes overwhelming.

To further compound the problem, latent defects are not static as development and testing proceeds. Old defects are continually being removed, new ones are creeping in, and the situation being modeled is continually changing. Even after system delivery, periodic software modifications may introduce new latent defects.

5.2.5 RMA Considerations for Systems Using COTS or NDI Hardware Elements

The desire to use COTS hardware as an alternative to custom-developed hardware for FAA systems means that the Government is unlikely to be able to exercise control over the internal design characteristics of the basic hardware elements used to construct systems. Both the reliability and maintainability of the elements are predetermined and largely beyond the control of the FAA. The only real option for the Agency is to require field data to substantiate a contractor's claims for the reliability and maintainability of their products.

The FAA's ability to influence the design of systems employing COTS/NDI components is primarily limited to demanding the removal and replacement of some unwanted hardware elements from their mountings, and possibly to requiring that hardware be built to industrial, instead of commercial standards.

5.3 The New Paradigm

The traditional RMA approach is not suitable for modern automation systems.

Paragraph 5.2 outlined how the technology advances, the characteristics of the systems being acquired, and the criticality of the applications in which these systems are used have changed over the last 40 years. It outlined several areas in which evolving changes have affected the traditional legalistic paradigm for requirements management. These changes have degraded the Government's ability to write and manage RMA requirements that satisfy the following three (out of ten) characteristics of good requirements cited in the System Engineering Manual (SEM):

- Allocatable
- Attainable (achievable or feasible)
- Verifiable

This handbook describes a new paradigm for RMA requirements management that focuses on applying NAS-Level requirements to tangible, physical Service Threads to assign them requirements that are achievable, verifiable, and consistent with the criticality of the service provided to users and specialists. The focus of the RMA management approach is on early identification and mitigation of technical risks affecting the performance of fault-tolerant systems, followed by an aggressive reliability growth program to provide contractual incentives to find and remove latent software defects.

The key elements of the approach are:

- Map the NAS-Level functional requirements to a set of generic Service Threads based on the NAPRS services reported for deployed systems. (Paragraph 6.2)
- Assign Service Thread Loss Severity Categories (STLSC) of "Safety-Critical," "Efficiency-Critical," or "Essential" to the Service Threads based on the effect of the loss of the Service Thread on NAS safety and efficiency of operations. (Paragraphs 6.3 and 6.4)
- Distinguish between *efficiency-critical* threads whose interruptions can be safety managed by reducing capacity that may, however, cause significant traffic disruption vs. *safety-critical* threads whose interruption could present a significant safety hazard during the transition to reduced capacity operations. (Paragraphs 6.3 and 6.4)
- Allocate NAS-Level availability requirements to Service Threads based on the criticality and associated availability requirements of the NAS capabilities supported by the threads.
- Recognize that the probability of achieving the availability requirements for any Service Thread identified as safety-critical is unacceptably low; therefore, decompose the thread into two independent new threads, each with a STLSC no greater than "efficiency-critical." The need for an independent backup thread for a safety-critical service needs to be recognized from the outset. Although there are no safety-critical threads in the field today, in many cases, backup systems have only been procured after a primary system failed to achieve the required availability.
- Recognize that the availability requirements associated with "efficiency-critical" Service Threads will require redundancy and fault tolerance to mask the effect of software failures.
- Move from using availability as a contractual requirement to parameters such as MTBF, MTTR recovery times, and mean time between successful recoveries – that is, testable requirements.

- Use RMA models only as a rough order of magnitude confirmation of the potential of the proposed hardware configuration to achieve the requirements, not a prediction of operational reliability.

- Focus RMA effort, during development, on design review and risk reduction testing activities to identify and resolve problem areas that could prevent the system from approaching its theoretical potential.

- Recognize that “pass/fail” reliability qualification tests are impractical for systems with high reliability requirements and substitute an aggressive reliability growth program.

- Use NAPRS data from the National Airspace System Performance Analysis System (NASPAS) on the RMA performance of currently fielded systems to assess the reasonableness and attainability of new requirements, and to verify that the requirements for new systems will result in systems with RMA characteristics that are at least as good as those of the systems they replace.

- Apply these principles throughout the acquisition process.

The application of these RMA management methods for the new approach is discussed in detail in Section 7. All phases of the acquisition process are addressed, including preliminary requirements analysis, allocation, preparation of procurement documents, proposal evaluation, contractor monitoring, and design qualification and acceptance testing.

6 DERIVATION OF NAS-LEVEL RMA REQUIREMENTS

This section presents background information on the methodology used to derive the NAS-Level Requirements. The primary purpose of defining NAS-level RMA Requirements is to relate the requirements for NAS Architecture Services and Capabilities to verifiable specifications for the hardware and software systems that will meet the user's expectations for the services provided by those systems. The NAS-SR-1000 document has been rewritten to align with the organization of the NAS Architecture Services and Capabilities. The original NAS-SR-1000 consisted of several hundred functional requirements. Each functional requirement was assigned a criticality rating of "Critical," "Essential," or "Routine." Each criticality rating was, in turn, associated with an availability requirement. During the rewrite of NAS-SR-1000, the functional requirements were essentially unchanged, except for "atomizing" them to conform to modern specification practices. Each of the atomized requirements was then assigned to one or more of the NAS Architecture Capabilities. The criticalities assigned to the functional requirements were unchanged.

The NAS-Level RMA requirements are not suitable to be incorporated directly into System-Level Specifications for acquisition programs. The NAS-SR-1000 functional requirements (e.g. provide aircraft position) or the broader NAS capabilities (e.g. aircraft-to-aircraft separation) involve both people and hardware, and are not easily related to RMA specifications for tangible systems to be procured by the FAA.

This handbook maps the NAS-SR-1000 functional requirements into Service Threads of interconnected systems that can be used by acquisition managers and Service Unit personnel to derive RMA requirements for System-Level Specifications.

There are five steps to this process:

1. Roll-up the criticalities associated with each NAS-SR-1000 functional requirement to the NAS Architecture Capability Level.
2. Map the NAS-SR-1000 capabilities to the Service Threads that support them.
3. Assess the contribution of each Service Thread to supporting the NAS-SR-1000 capability.
4. Based on that assessment, assign a Service Thread Loss Severity Category (STLSC) to the Service Thread.
5. Develop availability requirements associated with each STLSC.

Each of these steps is discussed in more detail below.

6.1 Roll-up NAS-SR-1000 Criticalities

The NAS-SR-1000 is organized around the NAS Architecture Services and Capabilities as defined by the current version of the NAS Architecture. These services and capabilities are listed in TABLE 6-1. There are nine NAS Architecture Services, designated by three digit numbers. Each Service provides two or more Capabilities, identified by four digit numbers. The nine Architecture Services provide a total of 23 Capabilities.

TABLE 6-1: NAS Architecture Services and Capabilities

Air Traffic Services/Capabilities
101 Flight Planning
1011 Flight Plan Support
1012 Flight Plan Processing (Flight Data Mgt.)
102 ATC-Separation Assurance
1021 Aircraft to Aircraft Separation
1022 Aircraft to Terrain/Obstacles Separation
1023 Aircraft to Airspace Separation
1024 Surface Separation
103 ATC-Advisory
1031 Weather Advisories
1032 Traffic Advisories
1033 NAS Status Advisories
104 Traffic Management – Synchronization
1041 Airborne Synchronization
1042 Surface Synchronization
105 Traffic Management-Strategic Flow
1051 Long Term Planning
1052 Flight Day Management
1053 Performance Assessment
106 Emergency and Alerting
1061 Emergency Assistance
1062 Alerting Support
107 Navigation
1071 Airborne Guidance
1072 Surface Guidance
108 Airspace Management
1081 Airspace Design
1082 Airspace for Special Use (Airspace Management)
109 Infrastructure/Information Management
1091 Monitoring and Maintenance
1092 Spectrum Management
1093 Government/Agency Support

6.1.1 Criticality Definitions

Associated with each Service/Capability are a number of individual functional and non-functional requirements. Earlier versions of the NAS-SR-1000 requirements had criticalities associated with each of these individual requirements. These definitions were based solely on the effect of the loss on the ability of the NAS to exercise safe separation and control over aircraft. A “critical” requirement was one for which “loss ***would prevent*** (emphasis added) the NAS from exercising safe separation and control over aircraft.”

The focus of these definitions on safety ignored one-half of the dual goals of the NAS mission to ensure the safe ***and orderly*** flow of air traffic (emphasis added). NAS systems and procedures have always been designed to maintain safety by implementing procedures to reduce capacity as required, and to avoid any situation that would ***prevent*** the NAS from exercising safe separation and control over aircraft.

The concept of associating NAS-SR-1000 criticalities (and associated availability requirements) with NAS-SR-1000 requirements is retained, but the level of assignment of criticalities is raised to the NAS Architecture Capability level and the criticality definitions are modified to reflect the dual goals of the NAS mission, (***safe*** and ***orderly*** flow of air traffic) and a risk-based orientation to the definitions has been substituted.

The dual premise behind this approach is that the NAS is safe and that the FAA goes to great lengths to maintain safety at all times. Systems are backed up with other systems, and when all else fails procedures are in place to ensure the safe operation of aircraft, albeit at some sacrifice of capacity and efficiency.

The revised definitions are:

1) **CRITICAL (.99999)** – Loss of this Service/Capability would raise the risk associated with providing safe and efficient local NAS operations, to an unacceptable level.

2) **ESSENTIAL (.999)** – Loss of this Service/Capability would significantly raise the risk associated with providing safe and efficient local NAS operations.

3) **ROUTINE (.99)** – Loss of this Service/Capability would have a minor impact on the risk associated with providing safe and efficient local NAS operations.

Note that the revised definitions retain the same availability requirements associated with the criticality levels as in the original NAS-SR-1000 document.

These revised definitions are based on the philosophy that there are no services whose loss would ***prevent*** the NAS from exercising safe separation and control of aircraft. Instead, loss of a system may increase the ***risk*** to an unacceptable level. There are procedures to cover any loss of surveillance, loss of communications, or even loss of both. Implementation of these procedures to cover such eventualities may severely disrupt the efficiency of NAS operations, but the most important objective of maintaining separation is preserved. Pilots also have a responsibility and are motivated to maintain separation. But the ***RISK*** of doing so at a busy ARTCC or TRACON without automation support is too high. Mitigating this risk leads to the requirement for high-reliability systems.

6.1.2 Criticality Roll-up

Criticalities associated with individual functional and non-functional requirements were rolled up to the NAS Architecture Capability level. The general rule was simply to examine all of the criticalities of the individual functional requirements contained under each NAS Architecture capability and then assign an overall criticality to the Capability based on the highest criticality of any of the individual constituent functional requirements contained in the Capability. However, mechanistically following this process could have led to unintended consequences. In general, all or most of the functional requirements assigned to a Capability have the same criticality, so the Capability simply inherited the criticality of its constituent requirements.

While performing the roll-up, the criticality assignment of each individual functional requirement was re-examined to make sure it was consistent within the context of the NAS Architecture Capability under consideration. The guiding principle was that the criticality assigned to an individual requirement must be realistic and constitute a significant factor in providing the overall NAS Architecture Capability. During the roll-up process, subjective adjustments were made to slightly more than one-half of the mechanical criticality roll-ups. There were three basic reasons for these exceptions to the general roll-up rule:

1) *All of the constituent functional requirements of the two Capabilities contained in the NAS Architecture Flight Planning Service were re-classified as "critical."*

Explanation: This exception followed from the decision to modify the original NAS-SR-1000 criticality definitions to incorporate efficiency as well as safety. Because loss of either Capability could have a critical impact on NAS efficiency, even if safety was not affected, both were increased from "essential" to "critical."

2) *The presence of a tiny fraction of "critical" requirements in a Capability was not allowed to drive the overall criticality above "essential."*

   Explanation: As a consequence of the mechanics of theNAS-SR-1000 rewrite process, a single functional requirement may appear under more than one NAS Architecture Capability. This can result in a situation where a requirement assigned a rating of "critical" may play a critical role in the context of one Capability, but not another. In the relatively few cases where a Capability contains hundreds of "essential" requirements and only one or two "critical" requirements, the overall rating was left at "essential." Examples can be found in the Capabilities contained in the ATC Advisory Service and the Traffic Management Synchronization Service. These Capabilities include a limited number of functional requirements that are also contained in the Capabilities contained in the Separation Assurance Service. While these functional requirements may indeed be critical in the context of the capabilities comprising the Separation Assurance Service, they should not be allowed to drive the criticalities of the ATC Advisory Service Capabilities or the Traffic Management Synchronization Service Capabilities.

3) *In cases where functional requirements bore no relation to real-time services, criticalities assigned to these functions were not included in the criticality roll-up.*

   Example: Requirements exist for FAA facilities to comply with OSHA construction standards and requirements to provide training facilities. The assignment of availability requirements to these functions was clearly inappropriate.

After applying the general roll-up rule and considering the special cases outlined above the results of the criticality roll-up appear in the NAS-SR-1000 roll-up column in the matrices in **FIGURE 6-7**, FIGURE 6-8FIGURE 6-8 and FIGURE 6-9 at the end of this section.

Another consideration in the roll-up process involves the fact that NAS-SR-1000 functional requirements are not allocated to specific systems. The roll-up process presupposes that all of the functional requirements comprising a Capability within a NAS Architecture Service will eventually be allocated to the same system. In this case, any lower criticality functional requirements contained within a Capability of a higher criticality simply inherit the higher criticality of the Capability and the system to which the requirements of that Capability are allocated.

Consider the case where a NAS Architecture Capability includes requirements that are almost certain to be allocated to different systems, for example, surveillance functions and weather functions. Should availability requirements for a system be driven by the consequence of one of its functional requirements being included in a critical Capability? This is at the heart of the conundrum of attempting to drive the availability requirements for real systems from NAS Services with unallocated functional requirements. The challenge is to devise a method for mapping the unallocated requirements associated with the NAS Architecture to something more tangible that can be related to specifications for real systems.

The method proposed is to relate the NAS Architecture Capabilities to a set of real-world services that are based on the services monitored by the National Airspace Performance Reporting System (NAPRS), as defined in FAA Order 6040.15D. To distinguish these NAPRS-based services from the NAS Architecture Services, they are designated as *Service Threads*, as discussed in the next section.

6.2 Map FAA Order 6040.15D Services to NAS-SR-1000 Service Threads

NAS Architecture Capabilities are supported by one or more Service Threads providing services to user/specialists (i.e., pilots and controllers). One example is surveillance data derived from sensors, processed into tracks, to which relevant data is associated and displayed to controllers. Another Service Thread is navigation data delivered to pilots. Service Threads are realized from interconnected facilities and systems.

The FAA's Air Traffic and Airways Facilities organizations have for years monitored a set of "Service Threads" under the NAPRS. NAPRS tracks the operational availability of what it calls, "services" (not to be confused with the NAS Architecture's Services and Capabilities). NAPRS tracks the operational availability and other RMA characteristics of services delivered by individual Service Threads to specialists. Because, in effect, these services represent a "contract" between airways facilities and air traffic, NAPRS services do not include NAS services, such as navigation, that are not used by air traffic controllers. (The performance of navigation *facilities* is monitored, but no corresponding service is defined.)

Basing the Service Threads on the NAPRS services defined in FAA Order 6040.15D provides several benefits. FAA personnel are familiar with the NAPRS services and they provide a common basis of understanding among Air Traffic Operations, Technical Operations and headquarters personnel. The operational data collected by NAPRS allows proposed RMA requirements to be compared with the performance of currently fielded systems to provide a check on the reasonableness of the requirements. The use of Service Threads permits the NAS architecture to evolve as components within a thread are replaced without the need to change the thread itself.

To realize these benefits, the Service Threads used in the NAS-SR-1000 and in this document should correlate as closely as possible with the services defined in FAA Order 6040.15D. However, it has been necessary to define some additional Service Threads that are not presently included in the FAA Order. The NAPRS monitors the performance of operational systems, while the NAS-SR-1000 looks toward requirements for future systems. Accordingly, new Service Threads will need to be created from time to time as the NAS evolves. This process should be closely coordinated with future revisions to FAA Order 6040.15D.

This section provides the traceability between the NAPRS services defined in FAA Order 6040.15D and the current list of Service Threads used in NAS-SR-1000 and in this handbook.

6.2.1 Taxonomy of FAA Systems

FAA systems used to provide the capabilities specified in NAS-SR-1000 can be divided into three major categories: Information Systems, Remote/Distributed elements, and Support Systems. FIGURE 6-1 presents a proposed taxonomy on which definitions and requirements allocation methodologies for the various categories of systems can be based. Strategies for each of these system categories are presented in the paragraphs that follow.

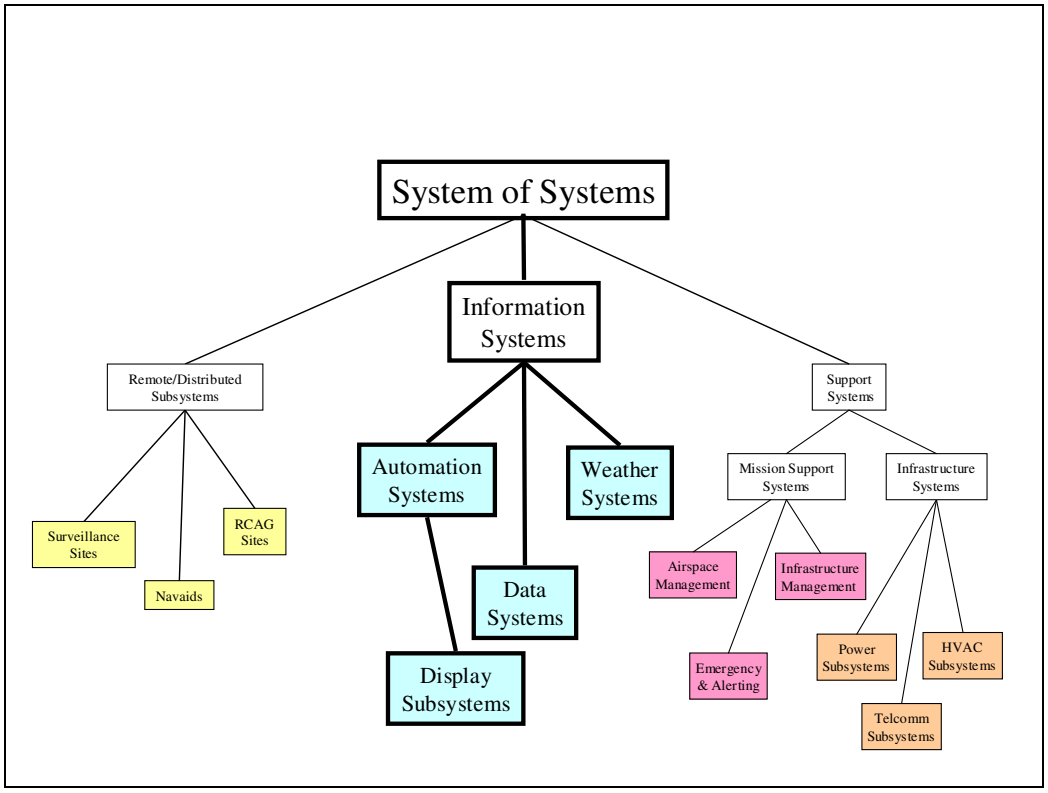

FIGURE 6-1 NAS System Taxonomy

Information systems are the primary focus of the requirements allocation methodology described in this Handbook and are outlined in bold in FIGURE 6-1. These systems are generally computer systems located in major facilities staffed by Air Traffic Control personnel. They consolidate large quantities of information for use by operational personnel in performing the NAS Air Traffic Control Mission. They usually have high criticality and availability requirements, because their failure could affect large volumes of information and many users. Typically, they employ fault tolerance, redundancy, and automatic fault detection and recovery to achieve high availability. These systems can be mapped to the NAS Services and Capabilities functional requirements.

The Remote/Distributed Subsystems category includes remote sensors, remote air-to-ground communications, interfacility data communications and navigation sites – as well as distributed subsystems such as display terminals – that may be located within a major facility. Failures of single elements, or even combinations of elements, can degrade performance at an operational facility, but generally they do not result in the total loss of the surveillance, communications, navigation, or display capability. Most of the Service Threads in the remote/distributed category are covered by the diversity techniques required by FAA Order 6000.36A, Communications Diversity.

Support systems include both Infrastructure Systems and Mission Support Systems. “Infrastructure” has become an overloaded term that can include a wide variety of systems. In this document, the scope of the systems included in the Infrastructure category is limited to those systems that provide power, environment, and basic communications services to the facilities that house the information systems. These systems can cause failures of the systems they support, so traditional allocation methods and the assumption of independence of failures do not apply to them.

The Mission Support category includes systems used to assist in managing the design of NAS airspace and the utilization of the electromagnetic frequency spectrum. NAS Infrastructure Management has been

included in the mission support category instead the Infrastructure category because the NAS infrastructure, as in the NAS Infrastructure Management System (NIMS), refers to the entire array of sensors, communications, computer systems, etc. that are used to perform the NAS mission and does not fit the more restrictive definition of the infrastructure category used in the Taxonomy Diagram in FIGURE 6-1. The NAS-SR-1000 criticality definitions and associated availabilities are based on the real time air traffic control mission. There is no basis for allocating these requirements to Service Threads and systems that indirectly support the air traffic control mission but that are not directly involved in the control of air traffic.

6.2.2 Categorization NAPRS Services

The NAPRS services defined in FAA Order 6040.15D were mapped to service threads, categorized in accordance with the major categories shown in the taxonomy in Figure 6-1. The “Remote/Distributed” Service Threads represent services that are provided by remote sensor and voice communications sites. These services generally represent a “many-to-one” or a “one-to-many” relationship with the control site. Failure of one of these services may degrade operations, but overlapping coverage and diversity in the set of distributed sites allows communications or surveillance functions to be maintained. These classifications and the requirements derivation methods appropriate to each classification are discussed in detail in Paragraph 7.1.1.

TABLE 6-2 lists the services from FAA Order 6040.15D and shows the mapping of these services to the three categories of Service Threads established by the taxonomy. A number of new Service Threads that do not have a corresponding FAA Order 6040.15D service were created and are shown at the bottom of the table. In addition, two “new” Service Threads based on NAPRS Reportable *Facilities* instead of the NAPRS *Services* have been incorporated. Four of the NAPRS services (ERAD, ESEC, MLSS, and PCSS) are not mapped to Service Threads. ERAD and ESEC are the broadband radar signals from an en route radar to the common digitizer. These are internal services contained within an Air Route Surveillance Radar (ARSR) site and are not generally provided outside of the radar site. MLSS is the service relating to the Microwave Landing System that will not be applicable to future navigation systems. PCSS is the power conditioning service thread. It is not used because the RMA requirements for power distribution services are addressed using the methodology described in Paragraph 6.7.3.

The first column of TABLE 6-2 provides the names of each of the Services defined in FAA Order 6040.15D. A “(NAPRS FACILITY)” entry in this column indicates a NAPRS reportable facility that is used as the basis for a service thread where there is no comparable service defined in FAA Order 6040.15D. A “(NEW)” entry in this column indicates a newly created Service Thread that does not have a corresponding facility or service in FAA Order 6040.15D. The remaining columns indicate the category of the service thread (Information, Remote/Distributed, or Support) and the domain of the service thread (Terminal, En Route, or Other). NAPRS services that have not been mapped to a service thread are also identified in these columns. The figure numbers in column one reference the Service Thread diagram in Appendix E.

The revised NAS-SR-1000 RMA requirements development process has augmented the NAPRS services in TABLE 6-2 with some additional Service Threads to include those services that are part of NAS but not included in the list of NAPRS services. NAPRS services representing lower level services from remote inputs that are included in composite higher level services provided to user/specialists, have been mapped to the Remote/Distributed column in TABLE 6-2 because the overall availability of the distributed communications and surveillance architecture is addressed by a “bottom-up” application of diversity and overlapping coverage techniques as directed by FAA Order 6000.36A instead of a top-down mathematical allocation of NAS-SR-1000 availability requirements. This is a consequence of the complex set of criteria that can affect the number and placement of remote sensor and communication sites and the overall availability of communications and surveillance services within a facility’s airspace.

Many of these factors such as the effects of terrain, traffic patterns, and man-made obstacles are not easily quantified and must be separately considered on a case-by-case basis. It is not practical to attempt to establish a "one size fits all" set of requirements for services in the Remote/Distributed category.

TABLE 6-2 Mapping of NAPRS Services to Service Threads

FAA Order 6040.15D Services	Information Service Threads	Remote/ Distributed Service Threads	Support Systems Service Threads	Domain
ASDES Airport Surface Detection Equipment Service [FIGURE E - 1]	ASDES			Terminal
AWPC Aviation Wx Processor/Concentrator [FIGURE E - 2]	AWPC			Other
AWPI Aviation Wx. Processor I/F [FIGURE E - 3]	AWPI			Other
AWPS Aviation Wx Processor Service [FIGURE E - 4]	AWPS			Other
AWPTE Aviation Wx. Processor Xfer – East [FIGURE E - 5]		AWPTE		Other
AWPTW Aviation Wx. Processor Xfer – West [FIGURE E - 5]		AWPTW		Other
BDAT Beacon Data (Digitized) [FIGURE E - 6]		BDAT		En Route
BUECS Backup Emergency Communications Service [FIGURE E - 7]		BUECS		En Route
CFAD Composite Flight Data Proc. [FIGURE E - 8]	CFAD			En Route
CFCS Central Flow Control Service [FIGURE E - 9]	CFCS			Other
CODAP Composite Oceanic Display and Planning [FIGURE E - 10]	CODAP			En Route
COFAD Composite Offshore Flight Data [FIGURE E - 11]	COFAD			En Route
CRAD Composite Radar Data Proc. [FIGURE E - 12]	CRAD			En Route
CTAS Center TRACON Automation System [FIGURE E - 13]	CTAS			En Route & Terminal
DRAD DARC Radar Data Proc. [FIGURE E - 14]	DRAD			En Route
ECOM En Route Communications [FIGURE E - 15]		ECOM		En Route
ERAD En Route Radar (Broadband)	(NOT MAPPED TO SERVICE THREAD)			
ESEC En Route Secondary Radar (Broadband)	(NOT MAPPED TO SERVICE THREAD)			
ETARS En Route Terminal Automated Radar Service [FIGURE E - 16]	ETARS			En Route
ETMS Enhanced Traffic Mgt. System [FIGURE E - 17]	ETMS			En Route
FCOM Flight Service Station Communications [FIGURE E - 18]		FCOM		Other
FDAT Flight Data Entry and Printout [FIGURE E - 19]		FDAT		En Route

TABLE 6-2 Mapping of NAPRS Services to Service Threads

FAA Order 6040.15D Services	Information Service Threads	Remote/ Distributed Service Threads	Support Systems Service Threads	Domain
FSSAS Flight Service Station Automated Service [FIGURE E - 20]		FSSAS		En Route
FSSPS Flight Service Station Processing Service [FIGURE E - 21]	FSSPS			En Route
IDAT Interfacility Data Service [FIGURE E - 22]		IDAT		En Route
LLWS Low Level Wind Service [FIGURE E - 23]	LLWS			Terminal
MDAT Mode S Data Link Data Service [FIGURE E - 24]		MDAT		Terminal & En Route
MLSS Microwave Landing System	(NOT MAPPED TO SERVICE THREAD)			
MPSS Maintenance Processor System Service [FIGURE E - 25]	MPSS			En Route
MSEC Mode S Secondary Radar Service [FIGURE E - 26]		MSEC		Terminal & En Route
NADS NADIN Switch [FIGURE E - 27]	NADS			Other
NAMS NADIN Message Transfer Switch [FIGURE E - 27]		NAMS		Other
NDAT NADIN Data Interchange Service [FIGURE E - 27]	NDAT			Other
PCSS Power Conditioning System Service (En Route)	(NOT MAPPED TO SERVICE THREAD)			
RDAT Radar Data (Digitized) [FIGURE E - 28]		RDAT		En Route
RTADS Remote Tower Alphanumeric Display Service [FIGURE E - 29		RTADS		Terminal
RTDRS Remote Tower Radar Display Service [FIGURE E - 30]		RTDRS		Terminal
RVRS Runway Visual Range Service [FIGURE E - 31]	RVRS			Terminal
TARS Terminal Automated Radar Service [FIGURE E - 32]	TARS			Terminal
TCOM Terminal Communications [FIGURE E - 33]		TCOM		Terminal
TRAD Terminal Radar [FIGURE E - 34]		TRAD		Terminal
TSEC Terminal Secondary Radar [FIGURE E - 35]		TSEC		Terminal
TDWRS Terminal Doppler Wx Radar Service [FIGURE E - 36]	TDWRS			Terminal
VSCSS Voice Switching and Control System Service [FIGURE E - 37]	VSCSS			En Route
WDAT WMSC Data Service [FIGURE E - 38] & FIGURE E - 40	WDAT			Other
WMSCS Weather Message Switching Service FIGURE E - 39 & FIGURE E - 40	WMSCS			Other

TABLE 6-2 Mapping of NAPRS Services to Service Threads

FAA Order 6040.15D Services	Information Service Threads	Remote/ Distributed Service Threads	Support Systems Service Threads	Domain
(NAPRS FACILITY) [FIGURE E - 41]	Terminal Voice Switch			Terminal
(NEW) [FIGURE E - 42]	Terminal Voice Switch Backup			Terminal
(NEW) [FIGURE E - 43	Terminal Surveillance Backup			Terminal
(NAPRS FACILITY) [FIGURE E - 44]	VTABS			En Route
(NEW) [FIGURE E - 45]	WAAS/GPS Service			En Route
(NEW) [FIGURE E - 46]	ADS/B Service			En Route
(NEW) [FIGURE E - 47]	Visual Guidance Service			Terminal
(NEW) [FIGURE E - 48]	RF Approach and Landing Services			Terminal
(NEW) [FIGURE E - 49]			NIMS	Support
(NEW) [FIGURE E - 50]		HF Voice Comm. Link		En Route
(NEW) [FIGURE E - 51]		RF Navigation Service		En Route
(NEW) [FIGURE E - 52]			Mission Services	Support

The newly created Service Threads at the bottom of the list are provided to serve as potential examples of new Service Threads. The Service Thread list will continue to evolve as the NAS architecture evolves and the NAPRS list of reportable Services is updated.

TABLE 6-3 provides a numerical summary of the mapping between NAPRS services and the Service Threads used in this handbook and in NAS-SR-1000. The intent of this table is to assure that there have

been no inadvertent omissions in the construction of the matrices in **FIGURE 6-7**, FIGURE 6-8, and FIGURE 6-9. Encapsulating this data in a single table will facilitate additions, deletions, and modifications to the Service Thread list as the NAS architecture evolves and the NAPRS list of reportable services is updated.

TABLE 6-3: Summary of Mapping of FAA Order 6040.15D Services to Service Threads

Service Thread Tally	Information Service Threads	R/D Service Threads	Support Service Threads	Totals
6040.15D Services	26	20		46
6040.15D Facilities	2			2
New Service Threads	6	2	2	10
6040.15D Services Not Used	(2)	(2)		(4)
Totals	32	20	2	54

TABLE 6-4 shows the mapping of the 54 Service Threads defined in TABLE 6-2 to the Matrices shown in **FIGURE 6-7**, FIGURE 6-8, and FIGURE 6-9. The reason that the total number of Service Threads in the three matrices is greater than the total number of Service Threads in TABLE 6-2 is that three of the Service Threads (CTAS, MDAT, and MSEC) appear in both the Terminal and En Route Matrices and therefore are counted twice, making the total number of Service Threads in the matrices three greater than the number of Service Threads in TABLE 6-2.

TABLE 6-4: Summary of Mapping of Service Threads to STLSC Matrices

	Information Service Threads	R/D Service Threads	Support Service Threads	Totals
Terminal STLSC Matrix	11	7		18
En Route STLSC Matrix	14	11		25
“Other” STLSC Matrix	4	8	2	14

Totals	29	26	2	57

(Note: There is an apparent discrepancy in FAA Order 6040.15D. There are 43 reportable services listed in Paragraph 301 and 46 services listed in the reference guide in Appendix 1 of the order. Paragraph 301 does not include. CTAS, MLSS, and RVRS but these services *are* contained in the 46 services listed in the Appendix in FAA Order 6040.15D.)

The final set of Service Threads to be used as the basis for NAS-Level RMA requirements is presented in the three matrices illustrated in FIGURE 6-7, FIGURE 6-8, and FIGURE 6-9. These matrices relate the Service Threads for the Terminal, En Route, and “Other” domains to the NAS Architecture Capabilities in NAS-SR-1000. The following paragraphs describe the process and rationale for constructing the matrices.

Each Service Thread is defined by a verbal description and a diagram. Each Service Thread diagram will also specify the NAS Architecture Capabilities supported by the Service Thread. The complete set of Service Thread diagrams is contained in Appendix E. A sample diagram illustrating the Composite Flight Data Processing Service (CFAD) Service Thread is illustrated in FIGURE 6-2. (This thread also contains lower level Service Threads that represent the data inputs from individual remote sites.)

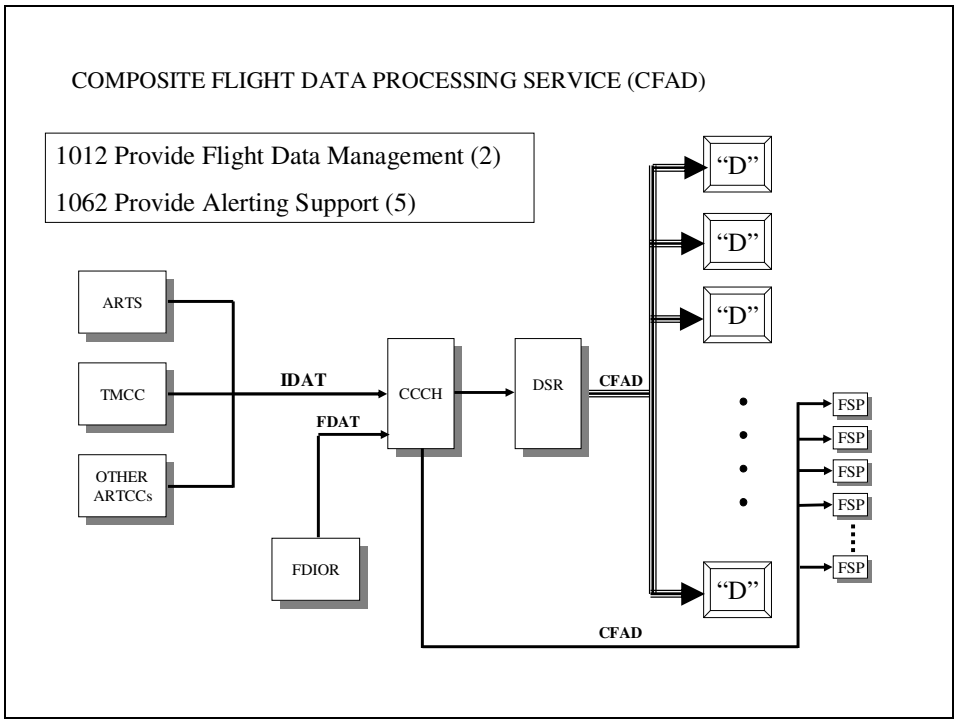

FIGURE 6-2: Example Thread Diagram (CFAD)

6.3 Assess Service Thread Contribution

To characterize the significance of the loss of a Service Thread, the revised NAS-SR-1000 requirements development process looked at the anatomy of a typical failure scenario.

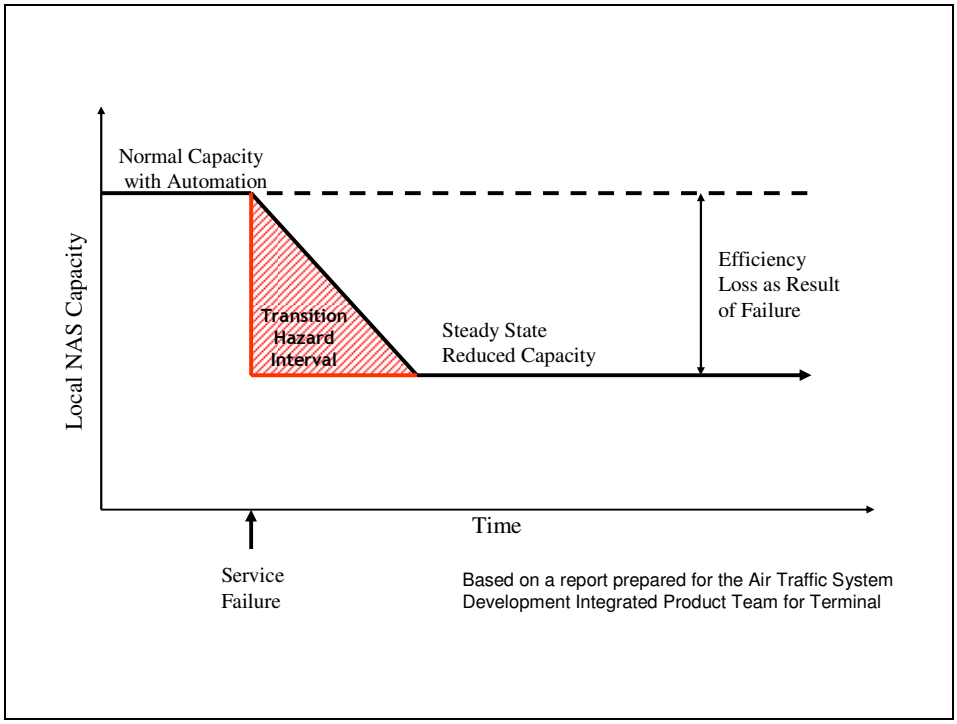

FIGURE 6-3: Effect of Service Interruptions on NAS Capacity

FIGURE 6-3 depicts the elements of a service failure scenario. Before the failure, with fully functional automation and supporting infrastructure, a certain level of local NAS capacity is achievable. After the failure, a hazard period exists while the capacity is reduced to maintain safety.

The potential effect of reducing capacity on efficiency depends on the level of demand. If the demand remains far below the airspace capacity for the available level of automation, then, whether or not it is reduced, there is no effect on efficiency. Trouble begins when the demand is close to the available capacity. If implementing procedures to accommodate a Service Thread failure causes the demand to exceed the available capacity, then queues start to build, and efficiency is impacted. The reduced capacity may be local, but the effects could propagate regionally or nationwide. The result is a potential loss of system efficiency with significant economic consequences as flights are delayed and cancelled.

Now, consider a critical NAS capability, such as flight plan processing supported by a Service Thread "A" (See FIGURE 6-4). The effect of the loss of a Service Thread on NAS safety and efficiency is characterized by a new term, "Service Thread Loss Severity Category" (STLSC—pronounced "Still See").

In Case 1, when the Service Thread fails, the controller switches to manual procedures that reduce traffic flow and increase separation to maintain safety. Depending on the level of demand at that Local NAS facility, the transition hazard period may be traversed without compromising safety. However, when the level of demand at that local facility is significant, the loss of the Service Thread may have safety implications and a significant ripple effect on the broader NAS. If it does, the Service Thread is assigned a Loss Severity Category of ***efficiency-critical***. Because the loss of an efficiency-critical Service Thread has regional or nation-wide impact, it might receive much attention and be disruptive, but not life threatening.

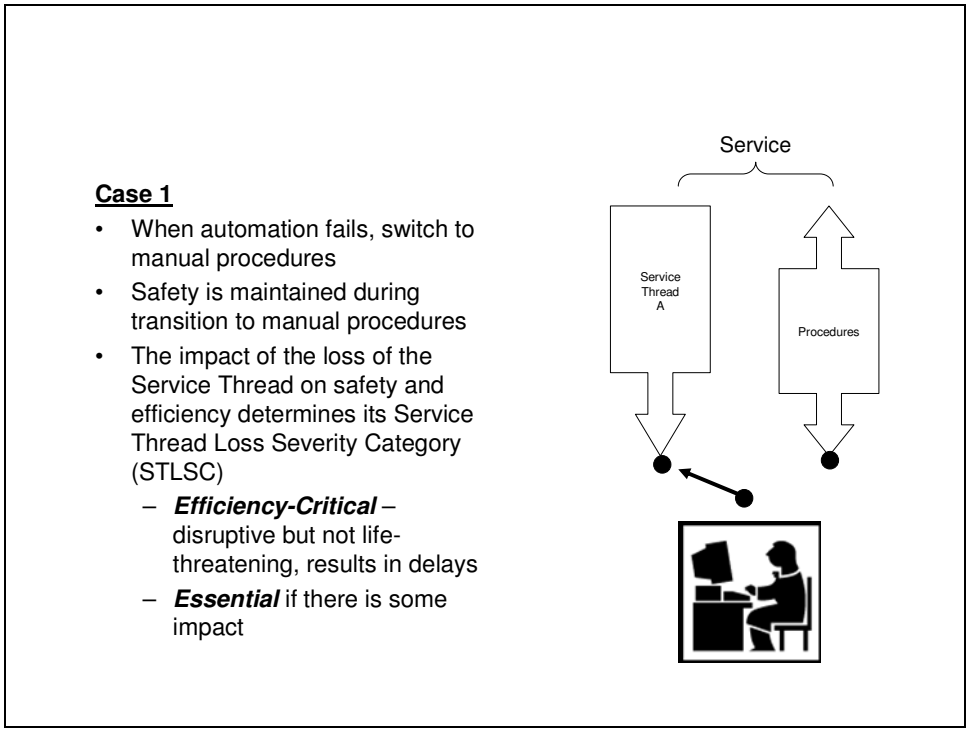

FIGURE 6-4: Service Thread Loss Severity Categories – Case 1

If loss of the Service Thread "A" has only localized impact, then it is considered to be of Loss Severity Category ***essential***. Loss of the Service Thread has an impact on local NAS efficiency. Such a loss is probably not newsworthy.

Now consider Case 2 (See FIGURE 6-5) – again a NAS service, such as aircraft-to-aircraft separation, for which a ***proposal*** exists to support it with a single Service Thread "X". The level of demand at the Local NAS facility, though, is such that the transition hazard period cannot be traversed without compromising safety. This is a potentially ***safety-critical*** situation that should not be, *and is not,* supported today by a single Service Thread. Loss of such a "safety-critical" Service Thread would likely result in a significant safety risk and increased controller stress levels during the transition to reduced capacity operations.

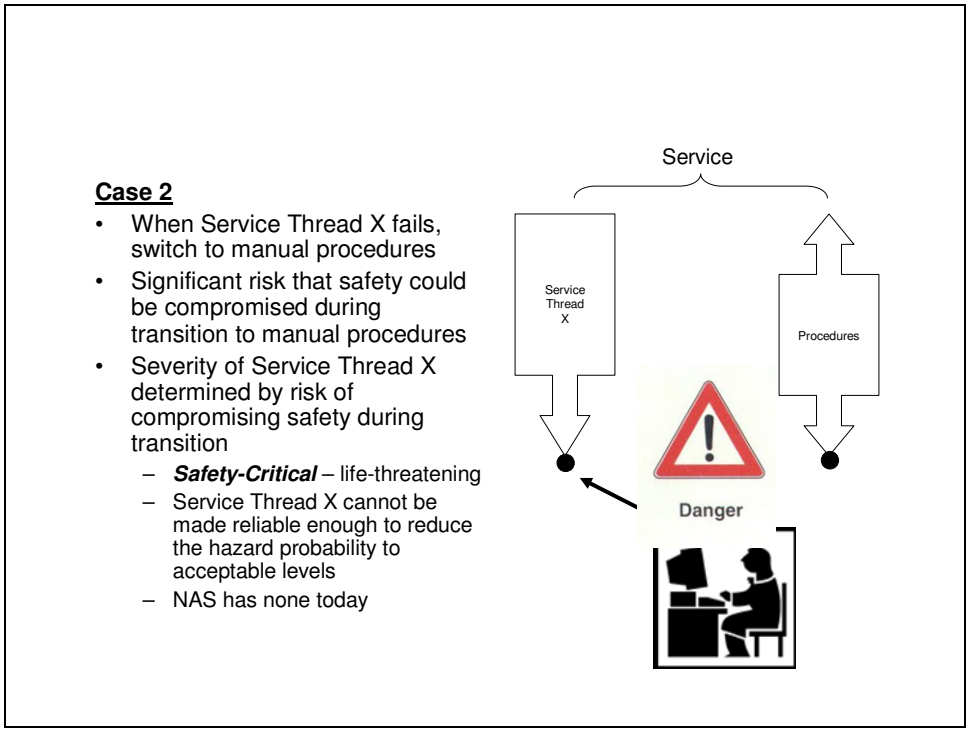

FIGURE 6-5: Potential Safety Critical Service Thread – Case 2

Note, "Safety-critical" relates to an assessment of the degree of hazard involved in the transition to a lower Local NAS Capacity. This designation distinguishes this set of circumstances from the more common safety analysis methods intended to define a direct cause and effect relationship between a failure and its consequences – for example, loss of digital fly-by-wire will almost certainly result in the loss of the aircraft and/or life. In contrast, loss of a ***safety-critical*** Service Thread will put undue stress on controllers, may result in some violations of separation standards, and an increased risk of a serious incident, but there is no certainty that a serious incident will result from the interruption.

Establishing requirements to make a safety-critical Service Thread so reliable that it will "virtually never fail" is unrealistic given today's state-of-the-art in software-intensive systems. The level of reliability and availability that would be required to support a safety critical Service Thread cannot be predicted or verified with enough accuracy to be useful, and has never been achieved in the field. For these reasons, any such requirements are meaningless. The FAA has learned this in the past and ***has no safety-critical Service Threads in the field***.

Perhaps a hypothetical single Service Thread supporting en route surveillance would, then, be safety-critical. In the field, however, the surveillance service has been decomposed into ***two*** independent Service Threads, CRAD and DRAD, each of which is only efficiency-critical. Similarly, the communications switching service has been decomposed into two independent Service Threads, e.g., VSCS and VTABS. By providing two independent threads, the unachievable requirements for a single safety-critical thread are avoided[5].

[5] For an extensive discussion of redundancy and diversity applied to air traffic control systems see En Route Automation Redundancy Study Task, Final Report, March 2000.

FIGURE 6-6 (Case 3) applies a second Service Thread, "Y," to complement Service Thread "X". It supplies sufficient functionality to maintain safety during the hazard period traversal. In this case safety is maintained because the controller can switch to the alternate Service Thread. Sufficient capacity may be provided thereby to maintain the efficiency that minimizes impact on the orderly flow of the NAS.

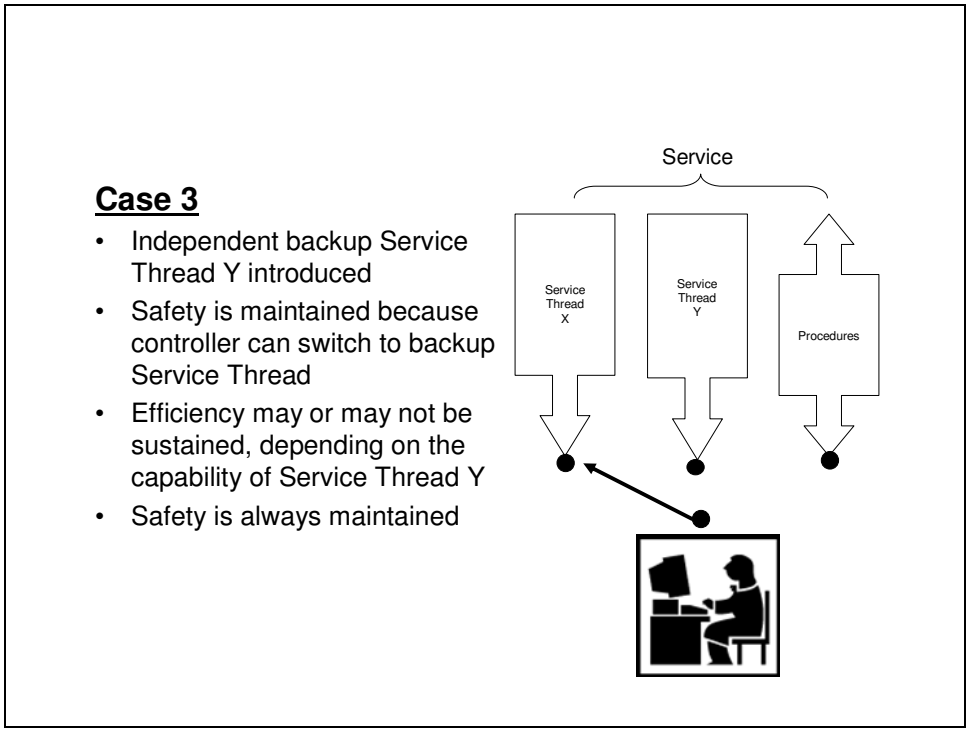

FIGURE 6-6: Decomposition of Safety-Critical Service into Threads

Whether or not the Service Thread needs to be full-service or reduced-capability depends on how much time is spent on the backup Service Thread.

The bottom line is – if a new Service Thread is determined to be "safety-critical," i.e. the transition to manual procedures presents a significant risk to safety, the potential new Service Thread must be divided into two independent Service Threads that can serve as primary and backup.

6.4 Assign Service Thread Loss Severity Category (STLSC)

In assessing the criticality of a Service Thread failure, there are two issues to consider:

- How hazardous is the transition to a reduced steady state capacity, e.g., is safety compromised while controllers increase separation and reduce traffic flow to achieve the reduced capacity state?

- What is the severity of the impact of the Service Thread failure on NAS efficiency and traffic flow? This severity depends on several non-system related factors such as the level of demand, level of the facility, time of day, and weather conditions.

If the transition risk is acceptable, the only issue in question is the effect of a failure on the efficiency of NAS operations. If the effect could cause widespread delays and flight cancellations, the Service Thread

is considered "**efficiency-critical**." If the reduction in NAS capacity results in some, but not widespread, disruptions to traffic flow, then the Service Thread is rated "**essential**." If, however, the hazard during transition to reduced traffic flow is significant, prudent steps must be taken to reduce the probability of that hazardous transition to an acceptable level. Experience has shown that this cannot be accomplished with a single Service Thread – instead a prudent design dictates use of two, independent Service Threads each designed to support the **safety-critical** requirement together with a simple manual capability for switching from one to the other (sometimes called a "knife switch" capability)[6].

[6] Redundancy alone is not enough to mitigate the effects of system faults. Redundant system components can provide continuing operations after a system fault only if the fault is not shared with the redundant resources performing the same function as the resource experiencing the fault. In the case of standby redundancy, the ability to provide continuing operations depends on the successful operation of complex automatic fault detection and recovery mechanisms. The beneficial effects of redundancy are therefore maximized when resources are organized into active redundant groups, and each group is made to be independent of the other groups. In this context, *independence* means that resources in one group do not rely on resources either contained in other groups or relied upon by the resources comprising other groups. Independence can apply to hardware or software.

FAA systems operate at extremely high availability (approximately .9999999), in large part because of the physical independence between the primary and secondary data path equipment, e.g., there is very little electrical and physical coupling between the Host and DARC systems, making it extremely rare for a Host failure to cause a DARC failure or vice versa. This is in contrast to the tight coupling of resources generally found between redundant resources within a data path. The sharing of resources between Host A and Host B, and between redundant DP and CP subsystems in the DARC, result in high availability for the Host and DARC systems (.9991 and.9999 respectively) by providing the capability to rapidly and non-disruptively recover from most failure modes.

This tight coupling induces a risk, however, that certain failure modes will result in the loss of all tightly coupled resources within a data path. A separate active data path loosely coupled to the failed data path is provided to ensure there is no disruption of service when these more comprehensive failure modes are encountered—providing in the case of the current system an overall availability of 0.9999999. A simple switching mechanism is also needed between the two data paths, which in the current system consists of a capability to switch nearly instantaneously between Host and DARC by depressing a single button. Therefore we conclude that to continue to provide the high availability of service that has been provided with the current system, redundant independent data paths are needed for the target architecture.

The most significant weakness of the HOST- DARC system is the reduced functionality available with the backup data path (DARC). Analysis of the NASPAS data shows that 98% of the time controllers are required to use the DARC system is to accommodate scheduled outages. This is essentially a procedural issue concerning Airways Facility (AF) technicians, although the close coupling of redundant Host processors is a significant factor. Whether or not the percentage of time spent on DARC to accommodate scheduled Host outages can be reduced, scheduled outages will always comprise a significant portion of total Host outages due to the desirability of retaining closely coupled resources within a data path to provide high availability, as discussed above. Therefore, to mitigate the effects of scheduled and unscheduled outages on full service availability, a full functionality secondary data path is required.

For these reasons, the target architecture should provide separate and independent full functionality continuously active data paths, with each data path being composed of tightly coupled redundant components. This will be required to ensure for the future the extremely high availability of equipment and services that has been achieved with the current system, and to mitigate the effects of scheduled outages better than is the case with the current system. The use of independent data paths is critical to achieving extremely high availability, and care must be taken not to compromise this characteristic during any phase of the system life cycle. Full functionality on both data paths is needed to mitigate the impacts of planned and unplanned primary channel outages, both of which are inevitable in an automation system.

This leads us to the following definitions for Service Thread Loss Severity Categories with comments on the defining characteristics of each:

1) **Safety Critical** – Service Thread loss would present an unacceptable safety hazard during transition to reduced capacity operations.

   - Loss of a Service Thread supporting the service would impact safety unless a simple, manual switchover to a backup Service Thread was successfully accomplished.
   - Depending on operational requirements, the secondary Service Thread might have a lower capacity than the primary.
   - FAA experience has shown that this capability is achievable if the service is delivered by two independent Service Threads, each built with off-the-shelf components in fault tolerant configurations.

2) **Efficiency Critical** – Service Thread loss could be accommodated by reducing capacity without compromising safety, but the resulting impact might have a localized or system-wide economic impact on NAS efficiency.

   - Experience has shown that this is achievable by a Service Thread built of off-the-shelf components in a fault tolerant configuration.

3) **Essential** – Service Thread loss could be accommodated by reducing capacity without compromising safety, with only a localized impact on NAS efficiency.

   - Experience has shown that this is achievable by a Service Thread built of good quality, industrial-grade, off-the-shelf components.

The revised NAS-SR-1000 RMA requirements development process has assigned a STLSC to each identified Service Thread. In terms of availability – as will be shown below – Service Thread Loss Severity Categories are one level of "nines" greater than the highest Service/Capability they support. Using this rule-of-thumb compensates for the fact that Service Threads may be composed of several systems, each with an availability no greater than that of the Service Thread itself.

6.5 Assigning Availability Requirements to STLSCs

In moving from the availability requirements associated with NAS criticality levels to STLSC availability requirements, the "Essential" requirement of .999 for NAS capabilities was increased to .9999 for a STLSC of "Essential." The availability requirement of .99999 associated with a "Critical" NAS capability was separated into an "Efficiency-Critical" STLSC requirement of .99999 and a "safety-critical" STLSC requirement of .9999999. Because the seven "nines" requirement is not achievable with a single Service Thread, a service that is potentially safety-critical must be supported by two independent "Efficiency-Critical" Service Threads.

The STLSC availability requirements were made greater than those associated with the criticalities of NAS architecture capabilities essentially to simplify the allocation process. Each NAS Architecture Capability is allocated to one or more Service Threads and each Service Thread contains one or more systems.

The equation for allocating availability is:

$$A_{Total} = A_1 \times A_2 \cdots\cdots A_n$$

(Where the A_i represents the availability allocated to an individual system.)

If the total availability is allocated equally across all systems, then:

$$A_{Subsystem} = (A_{Total})^{\frac{1}{n}}$$

(Where n represents the total number of systems.)

If n = 10, then the allocated availability required for each subsystem in a string will be an order of magnitude greater than the total availability. This relationship holds for any value of total availability. Rounding up the availability associated with the NAS Architecture Capabilities by one digit (i.e., nine) effectively allocates the total availability associated with a NAS Architecture Capability equally across ten subsystems. (For a more detailed discussion of this topic refer to Appendix C.)

Since the total number of subsystems in all of the threads supporting any given NAS Architecture Capabilities never reaches n = 10, then all availability allocations will be *less* than an order of magnitude greater than the total availability associated with a NAS Architecture Capability. Rounding up the availability by one "nine" provides enough of a margin to account for *both* multiple Service Threads supporting a given capability *and* the existence of multiple systems within the Service Threads.

This structure eliminates the necessity for FAA engineers to perform a mathematical allocation. It also eliminates the issue of whether the NAS-Level availability should be equally allocated across all systems in the thread, and it avoids the illusion of false precision that might stem from mathematical allocations. It is likely that any mathematical allocations would be rounded up to an even number of "nines" anyway. The risk, of course, of requiring that systems have availability an order of magnitude greater than the availabilities of the NAS Architecture Capabilities that they support is that the system availability requirement might be greater than absolutely necessary – and could conceivably cause systems to be more costly.

This should not be a problem for two reasons. First, the availabilities are associated only with the Service Threads that are directed toward computer systems. Other methods are proposed for Remote/Distributed elements and support systems. Secondly, a system designer is only required to show that the architecture's *inherent* availability meets the allocated requirement. The primary decision that needs to be made is whether the system needs to employ redundancy and automatic fault detection and recovery. With no redundancy, an inherent availability of three to four "nines" is achievable. With minimum redundancy, the inherent availability will realize a quantum increase to six to eight "nines." Therefore, allocated availabilities in the range of four to five "nines" will not drive the design. Any availability in this range will require redundancy and automatic fault detection and recovery that should easily exceed the allocated requirement of five "nines."

6.6 STLSC Matrix Development

The results of this process are summarized in the matrices in **FIGURE 6-7**, FIGURE 6-8, and FIGURE 6-9 that are provided as samples from NAS-SR-1000. (In the event of a discrepancy between these matrices and those in NAS-SR-1000, NAS-SR-1000 takes precedence.) These matrices provide the mapping between the NAS architecture capabilities in TABLE 6-1 and the Service Threads TABLE 6-2

The three matrices represent the Service Threads contained in each of the Terminal, En Route, and "Other" domains. All of the matrices contain Service Threads in both the Information and Remote/Distributed categories illustrated in FIGURE 6-1. In addition, the "Other" matrix contains the Service Threads in the Support Systems category of the taxonomy. Although the matrices are organized

around the domains of the Service Threads, the development of the RMA requirements for the Service Threads depends on their category in the taxonomy (Information, Remote/Distributed, or Support).

The column to the right of the NAS Services/Capabilities in the matrices represents the results of "rolling up" the criticalities assigned to each of the individual NAS-SR-1000 functional requirements contained in a NAS architecture capability to a single value representing the criticality of the entire NAS capability. Each NAS architecture capability is assigned a criticality level of "critical," "essential," or "routine." (The only capabilities having a "routine" criticality were those capabilities associated with non-real-time mission support. All of the capabilities associated with the real-time air traffic control mission have a criticality of "essential" or higher.)

Each matrix is divided into two sections. The top section above the black row contains information concerning the characteristics of the Service Threads, including the overall Service Thread Loss Severity Category (STLSC) for each Service Thread and the power requirements for the Service Thread. The section of the matrix below the black row shows the mapping between the NAS architecture capabilities and the Service Threads.

The individual cell entries in a *row* indicate which of the Service Threads support a given architecture capability. The individual cell entries in a Service Thread *column* indicate which of the various architecture capabilities are supported by the Service Thread. The numerical entries in the cells represent the Service Thread Loss Severity Category (STLSC) associated with the loss of a Service Thread on each of the specific architecture capabilities that are associated with that thread. A cell entry of "N" for "not rated" indicates one of two conditions: (1) Loss of the capability is overshadowed by the loss of a much more critical capability, which renders the provision of the capability meaningless in that instance, or (2) The capability is used very infrequently, and should not be treated as a driver for RMA requirements. For example loss of the ability to communicate with aircraft may affect the capability to provide NAS status advisories, but the affect of the loss of air-ground communications on the far more critical capability to maintain aircraft-to-aircraft separation overshadows the capability to provide NAS status advisories so it is "Not Rated" in this instance.

A column labeled "Manual Procedures" has been added on the right side of each of the matrices, with a "P" in every cell. This is to illustrate that the NAS capabilities are provided by FAA operational personnel using a combination of automation tools (Service Threads) and manual procedures. When service threads fail, manual procedures can still be employed to continue to provide the NAS capabilities supported by the failed Service Thread(s). The "P" in every cell indicates that there are always manual procedures to provide the NAS capabilities, so that the NAS-SR-1000 availability requirements associated with capability criticality can be achieved despite Service Thread interruptions.

The first row below the Service Thread names shows the pairing of Service Threads providing safety-critical services. The safety-critical service thread pairs are coded red and designated by an arrow spanning the two service threads. Note that the STLSC for each of the service threads making up a safety-critical pair is "efficiency-critical." This recognizes the fact that a *single* Service Thread is incapable of achieving the level of availability needed for safety-critical applications. The availability associated with each STLSC was obtained by rounding up the availabilities associated with NAS Architecture Capabilities as discussed in Paragraph 6.5.

The overall Service Thread Loss Severity Category (STLSC) for each Service Thread was obtained by "rolling up" the STLSCs for each of the cells in a Service Thread column. The overall STLSC for each Service Thread is represented by the highest criticality of any of the cells in the Service Thread's column. The overall STLSCs are in the second row below the Service Thread names.

The row(s) beneath these two rows represent the power system architectures associated with the Service Threads. Power distribution systems used by the service threads are discussed in greater detail in Paragraph 6.7.3.

Each of the matrices contains two categories of Service Threads: Service Threads representing *information services* provided to controllers by systems located within the facility, and Service Threads representing *Remote/Distributed Services* that include remote surveillance and communications sites serving the facility and intercommunications between the facility and other remote facilities.

As discussed in Paragraphs 6.2.1 and 6.2.2, only the Information Service Threads have STLSCs and availability requirements derived from NAS-SR-1000 associated with them. In contrast, the Remote/Distributed Service Threads are not associated with STLSCs and availability requirements, and use a "D" in the cells of the R/D Service Thread columns instead of a STLSC value to indicate which NAS capabilities are supported by the R/D Service Thread. The "D" is used to indicate that the diversity techniques in FAA Order 6000.36A instead of allocated availability from NAS-SR-1000 are used to achieve the required level of availability. It should be noted that an R/D Service Thread is a generic representation of a Service Thread with multiple instantiations. For example, in an ARTCC, the En Route Communications Service Thread (ECOM) will typically have several dozen instantiations of the Service Thread at specific locations (e.g. Atlantic City RCAG) to provide complete and overlapping coverage for the center's airspace.

The columns of the matrices have been color-coded to indicate which of the service threads are not directly mapped from the NAPRS services defined in FAA Order 6040.15D. Grey indicates newly created service threads that are not included in NAPRS and light green indicates service threads that are derived from NAPRS *facilities* as opposed to NAPRS *services* Light blue in an Information Service Thread name cell is used to indicate that the Service Thread is an "umbrella service" that drives several instantiations of a R/D Service Thread under the umbrella. Light blue in the entire column indicates those Service Threads that are under an umbrella Service Thread.

6.6.1 Terminal Systems STLSC Matrix

The matrix in **FIGURE 6-7** shows all Service Threads associated with the Terminal domain. These include both Information Service Threads and Remote/Distributed Service Threads. There are several rows corresponding to the Power System requirements for different size terminals as measured by their level of operations. Smaller facilities can have less stringent availability requirements and lower power capacity (KVA) requirements than the larger facilities. At smaller facilities manual procedures can be invoked to compensate for Service Thread interruptions without significantly disrupting traffic movement.

The Remote/Distributed Service Threads are characterized by equipment that is located at the control facility, (e.g. TRACON) and equipment that is remotely located and linked to the control facility by one or more communications paths. The equipment in the control facility is powered by the Critical bus of the Critical Power Distribution System. The remote equipment is powered by a separate power source. The last row above the black row, "Remote Site Power Architecture," specifies the power architecture requirements at the remote sites. In contrast with the Information Service Threads, Remote/Distributed Threads do ***not*** associate a quantitative STLSC availability requirement with each Service Thread.

The overall availability of the critical surveillance and communications services provided by these R/D Service Threads is achieved by employing diversity and redundancy techniques to circumvent failures of individual Service Thread instantiations. The diversity requirements for the Service Threads with a

STLSC rating of “D” are contained in FAA Order 6000.36A, *Communications Diversity*. Although these threads support critical NAS-SR-1000 capabilities, the required availability is achieved, not by a single Service Thread instantiation, but by a diverse architecture of distributed surveillance and communications sites with overlapping coverage.

NOTE: The newly created service thread “Terminal Voice Switch Backup” is not currently a physical backup system. Rather, it represents a capability and manual procedure for controllers to bypass a failed terminal voice switch by plugging directly into selected air/ground communications sites. Since this represents a workable, but significantly degraded communications capability, it is conceivable that a backup voice switch could be introduced at some point as the NAS architecture evolves. This is an example of how procedures can be used to assure continuity of services. The concept is particularly applicable to sites where the traffic density permits use of manual procedures to work around failed equipment without seriously affecting the safety and efficiency of operations.

The detailed methods for applying these requirements in the acquisition of new systems are described in Section 7.1.

Service/Capability -Terminal Service Thread STLSC Matrix

SR-1000 Capability Criticality Categories:
C = Critical = .99999
E = Essential = .999
R = Routine = .99

Service Thread Loss Severity Categories:
1 Safety-Critical = Paired Eff. Crit. Threads
2 Efficiency- Critical = .99999
3 Essential = .9999
D Addressed by Comm. Diversity Order
M Mission Support Services
P Manual Procedures
N (Not Rated)

Power System Architectures::
C2 CPDS Type 2
C1 CPDS Type 1
B BASIC
2A Comec'l Pwr+EG+UPS
1A Comec'l Pwr + EG + Mini UPS
U Comec'l Pwr+ UPS (no EG)
D Comec'l Pwr+Batteries
V Photovoltaic/Wind + Batteries
Z Independent Generation
1 Comec'l Pwr+EG
4 Comec'l Pwr
8 Dual Indep. Comm. Pwr.
S Same as Host Facility Power System Architecture
H = High Inherent Availability = .999998
R = Reduced Inherent Availability = .9998

	SR-1000 Roll-Up	Facility Power System Inherent Availability Requirement	Control Facility Service Threads: ASDES Airport Surface Detection Equipment Service	CTAS Center TRACON Automation System	LLWS Low Level Wind Service	RVRS Runway Visual Range Service	TARS Terminal Automated Radar Service	Emergency Surveillance Backup Service	TDWRS Terminal Doppler Wx Radar Service	TVS Terminal Voice Switch (Facility)	Terminal Voice Switch Backup (See Note in text)	Visual Guidance Facilities	RF Landing Aids Services	R/D Service Threads: MDAT Mode S Data Service	MSEC Mode S Secondary Radar	RTADS Remote Tower Alphanumeric Display Serv.	RTRDS Remote Tower Radar Display Service	TCOM Terminal Communications	TRAD Terminal Radar	TSEC Terminal Secondary Radar	Manual Procedures
Safety-Critical Thread Pairing																						
Service Thread Loss Severity Category			3	3	3	3	2	2	3	2	2	3	3	D	D	D	D	D	D	D		
Facility Power Architecture																						
Level 12 - Consolidated TRACON, Multiple Towers (e.g. PCT)		H	C2	C2	C2	C2	C2	C2	C2	C2	C2			C2	C2	C2	C2	C2	C2	C2		
Level 12 - Single Tower, collocated TRACON, Dual Beacon		H	C1	C1	C1	C1	C1	C1	C1	C1	C1			C1	C1	C1	C1	C1	C1	C1		
Level 12 - Single Tower, collocated TRACON, Single Beacon		H	B	B	B	B	B	B	B	B	B			B	B	B	B	B	B	B		
Level 12 - Single Tower, no TRACON (e.g. JFK)		R	2A	2A	2A	2A	2A	2A	2A	2A	2A			2A	2A	2A	2A	2A	2A	2A		
Level 11		H	C1	C1	C1	C1	C1	C1	C1	C1	C1			C1	C1	C1	C1	C1	C1	C1		
Levels 10, 9, & 8		H	B	B	B	B	B	B	B	B	B			B	B	B	B	B	B	B		
Levels 7 & 6		R	2A	2A	2A	2A	2A	2A	2A	2A	2A			2A	2A	2A	2A	2A	2A	2A		
Levels 5 & 4		R	1A	1A	1A	1A	1A	1A	1A	1A	1A			1A	1A	1A	1A	1A	1A	1A		
Levels 3 & 2		R	U	U	U	U	U	U	U	U	U			U	U	U	U	U	U	U		
Level 1		R	4	4	4	4	4	4	4	4	4			4	4	4	4	4	4	4		
Remote Site Power Architecture			A/1		D	1			1A			1/4	D	1A	1A	S	S	D	1A	1A		
Air Traffic Services																						
101 Provide Flight Planning Services																						
1011 Provide Flight Planning Support	C																					P
1012 Provide Flight Data Management	C																					P
102 Provide Separation Assurance																						
1021 Provide Aircraft to Aircraft Separation	C						2	2		2	2			D	D	D	D	D	D	D		P
1022 Provide Aircraft to Terrain/Obstacles Separation	C					3	2	2		2	2			D	D	D	D	D	D	D		P
1023 Provide Aircraft to Airspace Separation	C						2	2		2	2			D	D	D	D	D	D	D		P
1024 Provide [Aircraft/Vehicle] Surface Separation	C		3							2	2							D				P
103 Provide ATC Advisories																						
1031 Provide Weather Information	E				3		3	3	3	N	N					N	N	N	N			P
1032 Provide Traffic Advisories	E						3	3		N	N					N	N	N				P
1033 Provide NAS Status Advisories	E									N	N							N				P
104 Manage Traffic Synchronization																						
1041 Provide Airborne Traffic Management Synchronization	E			3			3	3		N	N					D	D	D	N	N		P
1042 Provide Surface Traffic Management Synchronization	E		3							N	N							D				P
105 Manage Strategic Traffic Flow																						
1051 Plan Long-Term	E																					P
1052 Manage Flight Day	E																					P
1053 Assess Performance	E																					P
106 Provide Emergency and Alerting Services																						
1061 Provide Emergency Assistance	R						N	N		N	N					N	N	N	N	N		P
1062 Provide Alerting Support	R																					P
107 Provide Navigation Services																						
1071 Provide Airborne Guidance	E											3	3									P
1072 Provide Surface Guidance	E											3										P
108 Manage Airspace																						
1081 Establish Airspace Design Criteria	R																					P
1082 Provide Airspace Design Management	R																					P
109 Manage NAS Infrastructure																						
1091 Monitor and Maintain the NAS	E																					P
1092 Manage Spectrum	R																					P
1093 Provide Government/Agency Support	R																					P
110 Other ATS Service Group Requirements	TBD																					

Notes:
ASDE-3 is Code 1A;
ASDE-X is Code 1

Runway Visual Range Reqts:
Code 4 for Cat I
Code 1 (CAT II TD)
Code 1 (CAT III) (TD,MP, RO)
Code 1 (CPA TD)
Code D (New Generation RVR)

Visual Guidance Facilities
Code 4 (CAT I)
Code 1 (CPA

FIGURE 6-7 Service/Capability – Terminal Service Thread STLSC Matrix[7]

[7] This figure is provided as a sample only, refer to NAS-SR-1000 for the approved requirements.

6.6.2 En Route STLSC Matrix

The En Route STLSC matrix in FIGURE 6-8 is similar to the Terminal STLSC matrix in **FIGURE 6-7**, except that it contains only a single row for the Control Facility Power System Architecture. ARTCCs do not have the wide range of activity levels that characterize the terminal domain, so a single Power System Architecture is used for all of the ARTCCs.

The En Route STLSC matrix introduces the NAPRS concept of "Umbrella Services." An umbrella service is a "one-to-many" service that supports numerous underlying facilities and services. An example of an umbrella service is the Flight Service Station Processing Service (FSSPS). The FSSPS Service Thread provides the processed flight service data to a number of Automated Flight Service Station (AFSS) facilities. Under the FSSPS umbrella are a number of instantiations of the Flight Service Station Automated Service (FSSAS) Service Thread that represents the transfer and display of flight service data at each remote AFSS facility. When FSSPS is interrupted, FSSAS is interrupted at all remote AFSS facilities.

The matrix includes four different Remote/Distributed Service Threads that are supported by three "umbrella services" that are higher in the NAPRS hierarchy. When an umbrella service fails, all of the associated services supported under the umbrella fail. The umbrella Service Threads are designated by a light blue background in the Service Thread name cell. R/D Service Threads under an umbrella Service Thread are designated by a light blue background for the entire Service Thread column.

As in the Terminal STLSC matrix, the En Route matrix includes a row for the power system architectures at remote sites that are powered separately from the control facility power system.

The detailed methods for applying these requirements in the acquisition of new systems are described in Section 7.1.

Service/Capability En Route Service Thread STLSC Matrix

SR-1000 Capability Criticality Categories:
C = Critical = .99999
E = Essential = .999
R = Routine = .99

Service Thread Loss Severity Categories:
1 Safety-Critical = Paired Eff. Crit. Threads
2 Efficiency- Critical = .99999
3 Essential = .9999
D Addressed by Comm. Diversity Order
M Mission Support Services
P Manual Procedures
N (Not Rated)

Power System Architectures::
C2 CPDS Type 2
C1 CPDS Type 1
B BASIC
2A Comec'l Pwr+EG+UPS
1A Comec'l Pwr + EG + Mini UPS
U Comec'l Pwr+ UPS (no EG)
D Comec'l Pwr+Batteries
V Photovoltaic/Wind + Batteries
Z Independent Generation
1 Comec'l Pwr+EG
4 Comec'l Pwr
8 Dual Indep. Comm. Pwr.
S Same as Host Facility Power System Architecture
H = High Inherent Availability = .999998
R = Reduced Inherent Availability = .9998

	SR-1000 Roll-Up	Facility Power System Inherent Availability Requirement	Control Facility Service Threads: CFAD Composite Flight Data Proc. (Umbrella Service)	CODAP Composite Oceanic Display and Planning	COFAD Composite Offshore Flight Data	CRAD Composite Radar Data Proc.	DRAD DARC Radar Data Proc.	CTAS Center/TRACON Automation System	ETARS En Route Terminal Automated Radar Serv.	ETMS Enhanced Traffic Mgt. System	FSSPS Flight Service Station Processing Service (Umbrella Service)	MPSS Maintenance Pprocessor Subsystem Service	VSCSS Voice Switching and Control System Service (Umbrella Service)	VTABS VSCS Training and Backup Switch (Facility)	WAAS/GPS Service	ADS-B Service	R/D Service Threads: ARINC HF Voice Communications Link	RDAT Radar Data (Digitized)	BDAT Beacon Data (Digitized)	BUECS Backup Emegency Communications Serv	ECOM En Route Communications (Under VSCSS Umbrella)	Terrestrial RF-based Navigation Systems	FDAT Flight Data Entry and Printout (Under CFAD Umbrella)	FSSAS Flight Service Station Automated Service (Under FSSPS Umbrella)	IDAT Interfacility Data Service (Under CFAD Umbrella)	MDAT Mode S Data Service	MSEC Mode S Secondary Radar	Manual Procedures
Safety-Critical Thread Pairing						↔	↔						↔	↔														
Service Thread Loss Severity Category			2	2	2	2	2	3	3	3	3	3	2	2	3	3	D	D	D	D	D	D	D	D	D			
Control Facility Power System Architecture		**H**	C2	C2	C2	C2	C2	C2	C2	C2	C2	C2	C2	C2	C2	C2	?	C2	C2	C2	C2		C2	C2	C2			
Remote Site Power System Architecture																	?	1	1	*	D/1	D	S	1A	S			
Air Traffic Services																												
101 Provide Flight Planning Services																												
1011 Provide Flight Planning Support	C										3						D							D				P
1012 Provide Flight Data Management	C		2	2	2						3						D						D	D	D			P
102 Provide Separation Assurance																												
1021 Provide Aircraft to Aircraft Separation	C					2	2		3				2	2		3	D	D	D	D	D					D	D	P
1022 Provide Aircraft to Terrain/Obstacles Separation	C					2	2		3				2	2		3		D	D	D	D					D	D	P
1023 Provide Aircraft to Airspace Separation	C					2	2		3				2	2		3	D	D	D	D	D					D	D	P
1024 Provide [Aircraft/Vehicle] Surface Separation	C															3												P
103 Provide ATC Advisories																												
1031 Provide Weather Information	E					3	3		3		3		3	3						N	N			D				P
1032 Provide Traffic Advisories	E					3	3		3	3			3	3				N	N	N	N							P
1033 Provide NAS Status Advisories	E										3		3	3						N	N			D				P
104 Manage Traffic Synchronization																												
1041 Provide Airborne Traffic Management Synchronization	E					3	3	3		3			3	3		3			N	N	N				D			P
1042 Provide Surface Traffic Management Synchronization	E															3												P
105 Manage Strategic Traffic Flow																												
1051 Plan Long-Term	E																							D				P
1052 Manage Flight Day	E																							D				P
1053 Assess Performance	E																							D				P
106 Provide Emergency and Alerting Services																												
1061 Provide Emergency Assistance	R					N	N		N				N	N				N	N	N	N							P
1062 Provide Alerting Support	R		N								N												N	N				P
107 Provide Navigation Services																												
1071 Provide Airborne Guidance	E														3	3						D						P
1072 Provide Surface Guidance	E															3												P
108 Manage Airspace																												
1081 Establish Airspace Design Criteria	R																											P
1082 Provide Airspace Design Management	R																											P
109 Manage NAS Infrastructure																												
1091 Monitor and Maintain the NAS	E											3																P
1092 Manage Spectrum	R																											P
1093 Provide Government/Agency Support	R																											P
110 Other ATS Service Group Requirements	TBD																											

* Note: BUECS is Code 1 at sites with PSC 1; Code D at sites with PSC D and a compatible external battery source; PSC 4 at all other sites.

ECOM is Code 1 at sites with AC Linear Power Amplifier, and Code D at all other sites.

FIGURE 6-8 Service/Capability – En Route Service Thread STLSC Matrix[8]

[8] This figure is provided as a sample only, refer to NAS-SR-1000 for the approved requirements.

6.6.3 "Other" Service Thread STLSC Matrix

The "Other" Service Thread STLSC matrix in FIGURE 6-9 contains the Service Threads that are not in Terminal facilities or ARTCCs, such as Aviation Weather, Flight Service Station, and National Airspace Data Interchange Network (NADIN) Service Threads.

In addition, the matrix includes two newly created Service Threads representing Mission Support Services and the National Airspace System (NAS) Infrastructure Management System (NIMS).

The Mission Support Service Thread with a STLSC rating of "M" is a generic service thread that encompasses a wide variety of simulators, data base management systems and manual procedures used to manage the design of the NAS airspace and to monitor and maintain the systems used in the performance of the NAS air traffic control mission. These systems are, for the most part, not 24/7 real time systems, and, in any event, cannot be directly related to the NAS-SR-1000 criticality definitions relating to the safe and efficient control of air traffic. The RMA requirements for systems providing mission support services are not derived from NAS-SR-1000, but instead are established by acquisition managers, based on what is commercially available and life cycle cost considerations.

Similarly, the NIMS is a mission support system with a STLSC rating of "M" whose RMA requirements are not derived from NAS-SR-1000 real-time availability requirements.

The detailed methods for applying these requirements in the acquisition of new systems are described in Section 7.1.

Service/Capability "Other" Service Thread STLSC Matrix

SR-1000 Capability Criticality Categories:
C = Critical = .99999
E = Essential = .999
R = Routine = .99

Service Thread Loss Severity Categories:
1 Safety-Critical = Paired Eff. Crit. Threads
2 Efficiency- Critical = .99999
3 Essential = .9999
D Addressed by Comm. Diversity Order
M Mission Support Services
P Manual Procedures
N (Not Rated)

Power System Architectures::
C2 CPDS Type 2
C1 CPDS Type 1
B BASIC
2A Comec'l Pwr+EG+UPS
1A Comec'l Pwr + EG + Mini UPS
U Comec'l Pwr+ UPS (no EG)
D Comec'l Pwr+Batteries
V Photovoltaic/Wind + Batteries
Z Independent Generation
1 Comec'l Pwr+EG
4 Comec'l Pwr
8 Dual Indep. Comm. Pwr.
S Same as Host Facility Power System Architecture
H = High Inherent Availability = .999998
R = Reduced Inherent Availability = .9998

		Control Fac.									R/D Serv. Thrds.						
	SR-1000 Roll-Up	AWPC Aviation Wx Processor/Concentrator	AWPI Aviation Wx. Processor I/F	AWPS Aviation Wx Processor Service (Umbrella Service)	CFCS Central Flow Control Serv.	NADS NADIN Switch (Umbrella Service)	NDAT NADIN Data Interchange Service	WMSCS Weather Message Switching Service	WDAT WMSC Data Service	AWPTE Aviation Weather Processor Transfer - East (Under AWPS Umbrella)	AWPTW Aviation Weather Processor Transfer - West (Under AWPS Umbrella)	FCOM Flight Service Station Communications	NAMS NADIN Message Transfer Service (Under NADS Umbrella)	Manual Procedures		Mission Support Services	NIMS Service
Safety-Critical Thread Pairing																	
Service Thread Loss Severity Category		3	3	3	3	2	2	3	3	D	D	D	D			M	M
Control Facility Power System Architecture		2A	2A	2A	2A	2A	2A	1	2A	2A	2A	C2	2A			?	?
Remote Site Power System Architecture										S	S	S	S				
Air Traffic Services																	
101 Provide Flight Planning Services																	
1011 Provide Flight Planning Support	C					2	2						D	P			
1012 Provide Flight Data Management	C					2	2						D	P			
102 Provide Separation Assurance																	
1021 Provide Aircraft to Aircraft Separation	C													P			
1022 Provide Aircraft to Terrain/Obstacles Separation	C													P			
1023 Provide Aircraft to Airspace Separation	C													P			
1024 Provide [Aircraft/Vehicle] Surface Separation	C													P			
103 Provide ATC Advisories																	
1031 Provide Weather Information	E	3	3	3		3	3	3	3	D	D	D	D	P			
1032 Provide Traffic Advisories	E													P			
1033 Provide NAS Status Advisories	E	3	3	3				3	3	D	D	D		P			
104 Manage Traffic Synchronization																	
1041 Provide Airborne Traffic Management Synchronization	E													P			
1042 Provide Surface Traffic Management Synchronization	E													P			
105 Manage Strategic Traffic Flow																	
1051 Plan Long-Term	E				3									P			
1052 Manage Flight Day	E				3									P			
1053 Assess Performance	E				3									P			
106 Provide Emergency and Alerting Services																	
1061 Provide Emergency Assistance	R											N		P		M	
1062 Provide Alerting Support	R													P		M	
107 Provide Navigation Services																	
1071 Provide Airborne Guidance	E													P			
1072 Provide Surface Guidance	E													P			
108 Manage Airspace																	
1081 Establish Airspace Design Criteria	R													P		M	
1082 Provide Airspace Design Management	R													P		M	
109 Manage NAS Infrastructure																	
1091 Monitor and Maintain the NAS	E													P			M
1092 Manage Spectrum	R													P		M	
1093 Provide Government/Agency Support	R													P		M	
110 Other ATS Service Group Requirements	TBD																

FIGURE 6-9 Service/Capability – "Other" Service Thread STLSC Matrix[9]

[9] This figure is provided as a sample only, refer to NAS-SR-1000 for the approved requirements.

6.7 NAS-SR-1000 RMA Requirements

Availability is an operational performance measure (see Paragraph 5.2.3) that is not well suited to contractual requirements or specifications. MIL-STD-961E, the Department of Defense standard for the format and content of military specifications, precludes citing availability as a requirement together with measures of reliability and maintainability.

The primary uses of the availability requirements associated with the Service Threads are to:

- Compare architecture alternatives during preliminary requirements analysis,
- Identify the need for redundancy and fault tolerance, and
- Provide a criterion for assessing the initial acceptability of architectures proposed by contractors.

Because availability cannot be used as a direct performance measure for verification purposes, this handbook makes greater use of other measures. It relies, instead, on a combination of requirements for reliability, maintainability, and verifiable recovery times that accurately specify characteristics of the frequency and duration of service interruptions to user/specialists.

6.7.1 Information Systems

TABLE 6-5 presents the reliability, maintainability, and recovery times for each of the Information Service Threads shown in the matrices in **FIGURE 6-7**, FIGURE 6-8, and FIGURE 6-9. The maintainability (MTTR) is based on the Airway Facilities standard requirement of 30 minutes. Recovery times are specified for those Service Threads that are required to incorporate fault tolerance automatic recovery. Two values of MTBF are specified. The first value represents the mean time successful automatic recoveries that are performed within the prescribed recovery time. The second value is the mean time between service interruptions for which the restoration time exceeds the prescribed recovery time, either because of unsatisfactory operation of the automatic recovery mechanisms, or because human intervention is required to restore service. (For Service Threads that do not require automatic recovery, the automatic recovery time is “N/A” and both MTBF values are equal.)

TABLE 6-5: Service Thread Reliability, Maintainability, and Recovery Times[10]

Service Thread	Maintainability (MTTR) (hours)	Automatic Recovery Time (sec)	Reliability (MTBF)	
			Less than Automatic Recovery Time (hours)	Greater or equal to Automatic Recovery Time (hours)
Airport Surface Detection Equipment Service (ASDES)	0.5	N/A	5000	5000
Aviation Wx Processor /Concentrator (AWPC)	0.5	N/A	5000	5000
Aviation Wx. Processor I/F (AWPI)	0.5	N/A	5000	5000
Aviation Wx. Processor Service (AWPS)	0.5	N/A	5000	5000
Composite Flight Data Proc. (CFAD)	0.5	10	300	50,000
Central Flow Control Serv. (CFCS)	0.5	N/A	5000	5000
Composite Oceanic Display and Planning (CODAP)	0.5	10	300	50000
Composite Offshore Flight Data (COFAD)	0.5	10	300	50000
Composite Radar Data Proc. (CRAD)	0.5	10	300	50,000
Center TRACON Automation System (CTAS)	0.5	N/A	5000	5000
DARC Radar Data Proc. (DRAD)	0.5	10	300	50,000
En Route Terminal Automated Radar Serv. (ETARS)	0.5	5	300	50,000
Enhanced Traffic Mgt. System (ETMS)	0.5	N/A	5000	5000
Flight Service Station Processing Service (FSSPS)	0.5	N/A	5000	5000
Low Level Wind Service (LLWS)	0.5	N/A	5000	5000
Maintenance Processor System Service (MPSS)	0.5	N/A	5000	5000
NADIN Switch (NADS)	0.5	10	300	50,000
NDAT NADIN Data Interchange Service	0.5	10	300	50,000
Runway Visual Range Service (RVRS)	0.5	N/A	5000	5000
Terminal Automated Radar Service (TARS)	0.5	5	300	50,000

[10] This table is provided as a sample only, refer to NAS-SR-1000 for the approved requirements.

TABLE 6-5: Service Thread Reliability, Maintainability, and Recovery Times

Terminal Doppler Wx Radar Service (TDWRS)	0.5	N/A	5000	5000
Voice Switching and Control System Serv. (VSCSS)	0.5	10	300	50,000
VSCS Training and Backup Switch Service (VTABS)	0.5	10	300	50,000
WMSC Data Service (WDAT)	0.5	N/A	5000	5000
Weather Message Switching Service (WMSCS)	0.5	N/A	5000	5000
Visual Guidance Facilities	0.5	N/A	5000	5000
RF Guidance Facilities	0.5	N/A	5000	5000
WAAS/GPS Service	TBD			
ADS-B Service	TBD			
Terminal Surveillance Emergency Backup	.0.5	10	300	50000
Terminal Voice Switch (TVS)	0.5	10	300	50000
Terminal Voice Backup	TBD			
Efficiency Critical (New)	0.5	10	300	50,000
Essential (New)	0.5	N/A	5000	5000

6.7.2 Remote/Distributed Service Threads

Lacking any straightforward methodology for performing a quantitative top-down allocation of NAS-Level requirements for remote/distributed subsystems, NAS-SR-1000 imposes no RMA requirements on such systems. The RMA characteristics for these systems, therefore, are established primarily by life cycle cost and diversity considerations. The Remote/Distributed Service Threads are presented in TABLE 6-6. Diversity issues are a complex function of local traffic patterns, terrain, etc. This topic is discussed in greater detail in Paragraph 7.1.1.2.

TABLE 6-6: Remote/Distributed Service Threads

Remote/Distributed Service Threads	Control Facility	Remote Site	Service Type
AWPTE Aviation Wx. Processor Xfer – East [FIGURE E - 5]	Atlanta AWP	ARTCC FSDPS	Transfer of Service B Wx Data between AWP & ARTCC FSDPS
AWPTW Aviation Wx. Processor Xfer – West [FIGURE E - 5]	Salt Lake AWP	ARTCC, FSDPS	Transfer of Service B Wx Data between AWP & ARTCC FSDPS
BDAT Beacon Data (Digitized) [FIGURE E - 6]	ARTCC, TRACON	ARSR	Digitized Secondary Radar Reports
BUECS Backup Emergency Communications Service [FIGURE E - 7]	ARTCC	BUEC	En Route A/G Voice Comm.
ECOM En Route Communications [FIGURE E - 15]	ARTCC	RCAG	En Route A/G Voice Comm.
FCOM Flight Service Station Communications [FIGURE E - 18]	AFSS	AFSS, ATCT, VOR	AFSS A/G Voice Comm.
FDAT Flight Data Entry and Printout [FIGURE E - 19]	ARTCC	TRACON ATCT	Flight Plan Data Transfer
FSSAS Flight Service Station Automated Service [FIGURE E - 20]	ARTCC	AFSS	Flight Service Data Transfer
IDAT Interfacility Data Service [FIGURE E - 22]	ARTCC	ARTCC TRACON	Computer-to-Computer Data Transfer
MDAT Mode S Data Link Data Service [FIGURE E - 24]	ARTCC, TRACON	ARSR ASR	Mode S Data Link Reports from Radar Site
MSEC Mode S Secondary Radar Service [FIGURE E - 26	ARTCC, TRACON	ARSR ASR	Mode S Secondary Radar Reports
NAMS NADIN Message Transfer Switch [FIGURE E - 27]	NADIN Switching Center	ARTCC	Transfer of Message. Data between NADIN Switching Center & ARTCC NADIN Concentrator
RDAT Radar Data (Digitized) [FIGURE E - 28]	ARTCC, TRACON	ARSR	Digitized Primary Radar Reports
RTADS Remote Tower Alphanumeric Display Service [FIGURE E - 29	ATCT	TRACON	A/N Display in ATCT from Remote TRACON Source
RTDRS Remote Tower Radar Display Service [FIGURE E - 30]	ATCT	TRACON	Radar Display in ATCT from Remote TRACON Source
TCOM Terminal Communications [FIGURE E - 33]	TRACON, ATCT	RTR	Terminal A/G Voice Communications

TABLE 6-6: Remote/Distributed Service Threads

TRAD Terminal Radar [FIGURE E - 34]	TRACON	ASR	Primary Radar Reports
TSEC Terminal Secondary Radar [FIGURE E - 35]	TRACON	ASR	Secondary Radar Reports
HF Voice Communications Service	ARINC		Oceanic Communications
VHF Omnidirectional Range Navigation	N/A	VOR Site	RF Navigation Service

Distributed Service Threads

The minimum number of operational displays should be based upon the needs of the operational environment of the host system. i.e. (n-x). Appendix C provides a discussion on the equations and method to solve this problem. This handbook also provides an Excel spread sheet to do the Math. Please note that the RMA requirement is superseded by the operational need. An example would be as follows:

If the availability (AV) for each display is .99 and the site has 10 displays, what is the AV if two displays are lost when the state of all displays is operational and then a third display fails? The AV would be 0.9998838. Mean Time between Failure (MTBF) in losing the third display is 1489.9 hours, and the Mean Time to Repair (MTTR) two displays at the same time is .17 hours. The site would need to address the operational need against this predicated risk in assessing the needs of the site.

6.7.3 Infrastructure Systems (Power Systems)

The RMA requirements for power systems as defined in the STLSC matrices are based on the STLSCs of the threads they support as well as the traffic level of the facility in which they are installed. All ARTCCs have the same RMA requirements and the same power architecture. The inherent availability requirements for Critical Power Distribution Systems (CPDS) are derived from the NAS-SR-1000 availability requirements for critical NAS capabilities.

In the Terminal domain, there is a wide range of traffic levels between the largest facilities and the smallest facilities. At larger terminal facilities, the service thread loss severity is comparable to that of ARTCCs and the inherent availability requirements are the same. Loss of service threads resulting from power interruptions can have a critical effect on efficiency as operational personnel reduce capacity to maintain safe separation and could increase safety hazards to unacceptable levels during the transition to manual procedures.

The power system architecture codes used in the matrices were derived from FAA Order 6950.2D, *Electrical Power Policy Implementation at National Airspace System Facilities.* This order contains design standards and operating procedures for power systems to ensure power system availability consistent with the requirements for the service threads supported by the power services.

However at smaller terminal facilities, manual procedures can be invoked without a significant impact on either safety or efficiency. Accordingly, the inherent availability capability requirements for these facilities can be reduced from those applied to the larger facilities.

NAS-SR-1000 inherent availability requirements should in no way be interpreted to be an indication of the predicted operational performance of a CPDS. The primary purpose of these requirements is simply to establish whether a dual path redundant architecture is required or whether a less expensive radial CPDS architecture is adequate for smaller terminal facilities.

The NAS-SR-1000 inherent availability requirements are only applicable to the Critical Power Distribution Systems required for ARTCC's and the larger Terminal facilities. Availability models of the power system architectures for these systems have been shown to be consistent with the requirements

Facility Level	Inherent Availability
Level 12	A = .999998
Level 9 – 11	A = .999998
Level 5 – 8	A = .9998
Other	FAA Order 6950.2D

derived from the NAS-SR-1000 requirements.

TABLE 6-7: Power System Allocated Inherent Availability[11]

n order to meet the inherent availability requirements, dual path architectures have been employed. The power for Safety-Critical Service Thread pairs should be partitioned across the dual power paths such that failure of one power path will not cause the failure of both Service Threads in the Safety-Critical Service Thread pair.

For smaller facilities such as those using commercial power with a simple Engine Generator or battery backup, there is no allocation of NAS-SR-1000 RMA requirements. Although CPDS architectures can be tailored to meet inherent availability requirements through the application of redundancy, there is no such flexibility in simple single path architectures using Commercial off the Shelf (COTS) components. Accordingly, for these systems, only the configuration will be specified using the Power Source Codes defined in FAA Order 6950.2D; no NAS-SR-1000 allocated inherent availability requirements will be imposed on the acquisition of COTS power system components. The reliability and maintainability of these COTS components shall be in accordance with best commercial practices.

The matrices in FIGURE 6-7**,** FIGURE 6-8, and FIGURE 6-9 provide the power system architecture requirements for Service Thread equipment located in both control facilities and, where applicable, at remote sites.

The power system requirements are presented in (in which standard power system configurations meeting these requirements have been established). The standards for power systems are contained in FAA Order 6950.2D. The table indicates that the larger facilities require a dual path redundant CPDS architecture that is capable of meeting the .999998 inherent availability requirement. Smaller facilities can use a single path CPDS architecture capable of meeting .9998 inherent availability. The smallest facilities do not require a CPDS architecture and use the specified power system architecture code with no NAS-SR-1000 allocated availability requirement

[11] This table is provided for illustrative purposes only. Refer to the NAS-SR-1000 for the approved requirements.

[13] The interface standards between infrastructure systems and the systems they support is an area of concern. For example, if power glitches are causing computer system failures, should the power systems be made more stable or should the computer systems be made more tolerant? This tradeoff between automation and power systems characteristics is important and deserves further study; however it is considered outside the scope of this study.

7 ACQUISITION STRATEGIES AND GUIDANCE

Acquisition cycles can span multiple months or even years. Successful deployment of a complex, high-reliability system that meet the user's expectations for reliability, maintainability and availability is dependent on the definition, execution, and monitoring of a set of interrelated tasks. The first step is to derive from the NAS-SR-1000, the requirements for the specific system being acquired. Next, the RMA portions of the procurement package must be prepared and technically evaluated. Following that, a set of incremental activities intended to establish increasing levels of confidence that the system being designed built and tested meets those requirements run throughout the design and development phases of the system. Completing the cycle is an approach to monitoring performance in the field to determine whether the resulting system meets, or even exceeds, requirements over its lifetime. This information then forms a foundation for the specification of new or replacement systems.

FIGURE 7-1 depicts the relationship of the major activities of the recommended process. Each step is keyed to the section that describes the document to be produced. The following paragraphs describe each of these documents in more detail.

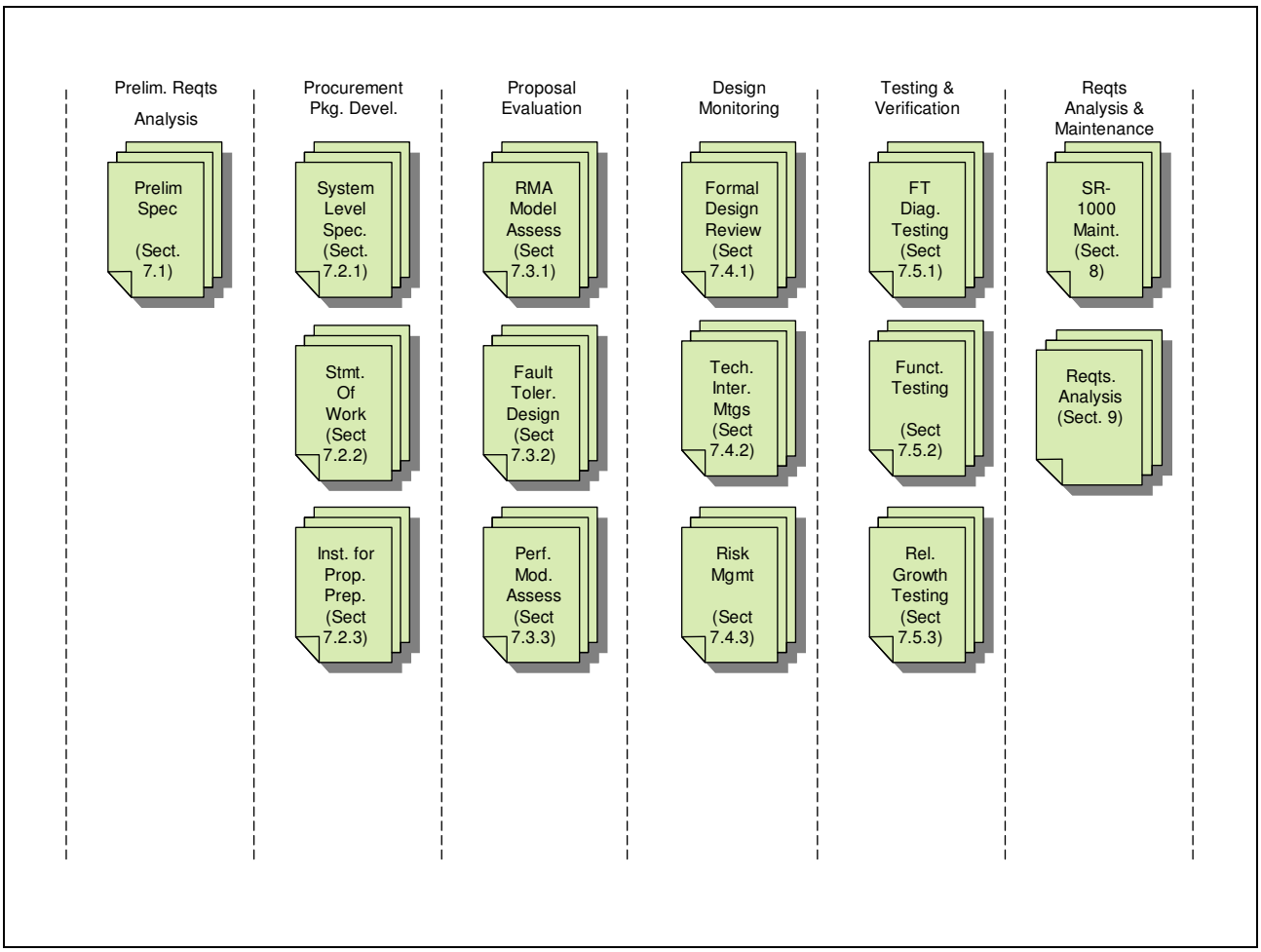

FIGURE 7-1: Acquisition Process Flow Diagram

7.1 Preliminary Requirements Analysis

This section presents the methodology to apply NAS-Level Requirements to major system acquisitions. The NAS-Level Requirements are analyzed to determine the RMA requirements allocated to the system and their potential implications on the basic architectural characteristics of the system to be acquired. The potential requirements are then compared with the measured performance of currently fielded systems to build confidence in the achievability of the proposed requirements and to ensure that newly acquired systems have RMA characteristics equal to, or better than, those of the systems they replace.

The first step in the process is to determine the category of the system being acquired: information systems, remote/distributed subsystems, or infrastructure systems. Each of these categories is treated differently, as discussed in the following section.

7.1.1 Taxonomy of FAA Systems and Associated Allocation Methods

There is no single allocation methodology that can logically be applied across all types of FAA systems. Allocations from NAS-Level requirements to the diverse FAA systems comprising the NAS require different methodologies for different system types. NAS systems are classified as falling into three major categories: Information Systems, Remote/Distributed subsystems, and Support Systems as discussed in Section 6.2.1. The taxonomy of FAA system classifications described in Section 6.2.1 and illustrated in FIGURE 6-1 is repeated in FIGURE 7-2. This taxonomy represents the basis on which definitions and allocation methodologies for the various categories of systems are established. Strategies for each of these system categories are presented in the paragraphs that follow.

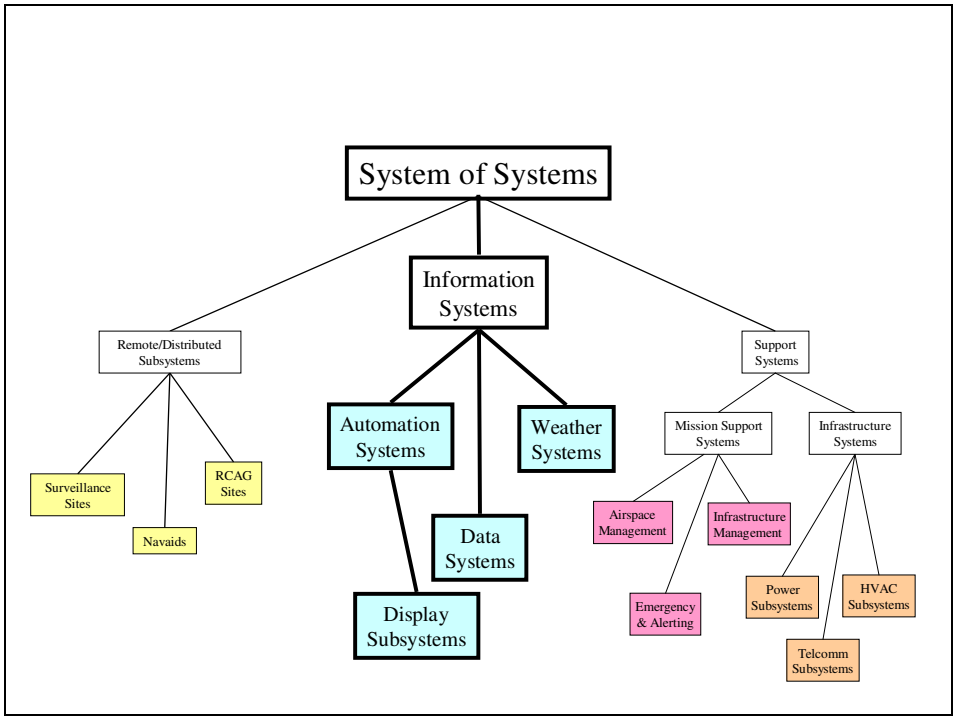

FIGURE 7-2: NAS System Taxonomy

Information Systems are generally computer systems located in major facilities staffed by Air Traffic Control personnel. These systems consolidate large quantities of information for use by operational

personnel. They usually have high criticality and availability requirements because their failure could affect large volumes of information and many users. Typically, they employ fault tolerance, redundancy, and automatic fault detection and recovery to achieve high availability. These systems can be mapped to the NAS Services and Capabilities functional requirements.

The Remote/Distributed Subsystems category includes remote sensors, communications, and navigation sites – as well as distributed subsystems such as display terminals – that may be located within a major facility. Failures of single elements, or even combinations of elements, can degrade performance at an operational facility, but generally they do not result in the total loss of the surveillance, communications, navigation, or display capability.

Support Systems include both Infrastructure Systems that provide power, environment, and basic communications services to the facilities that house the information systems and Mission Support Systems that assist in managing the design of the NAS airspace, and the operation and maintenance of the systems used in the performance of the air traffic control mission.

Only Service Threads in the Information Systems category have RMA requirements that are allocated from NAS-SR-1000 availability requirements.

Remote/Distributed Service Threads achieve the overall availability required by NAS-SR-1000 through the use of qualitative architectural diversity techniques as specified in FAA Order 6000.36A. Primarily, these involve multiple instantiations of the Service Thread with overlapping coverage. The ensemble of Service Thread instantiations provides overall continuity of service despite failures of individual Service Thread instantiations. There is no “top-down” quantitative allocation of NAS-SR-1000 availability requirements to R/D Service Threads. The RMA requirements for the systems and subsystems comprising R/D Service Threads are determined by Acquisition Managers in accordance what is achievable and Life Cycle Cost considerations.

Mission Support Service Threads do not have availability requirements allocated from NAS-SR-1000. The NAS-SR-1000 availability requirements are directed toward real-time air traffic control functions. These requirements are not applicable to tools used to manage the NAS airspace design and infrastructure. The RMA requirements for the systems and subsystems comprising Mission Support Service Threads are determined by Acquisition Managers in accordance what is achievable and Life Cycle Cost considerations.

Procedures for determining the RMA characteristics of the Power Systems supplying Service Threads are discussed in Paragraph 7.1.1.3.1

7.1.1.1 Information Systems

The starting point for the development of RMA requirements is the set of three matrices developed in the previous section, **FIGURE 6-7**, FIGURE 6-8, and, FIGURE 6-9

The first step in the process is to select the matrix pertaining to the domain in which a system is being upgraded or replaced and review the Service Threads listed in the matrix to determine which Service Thread(s) pertain to the system that is being upgraded or replaced.

For systems that are direct replacements for existing systems:

1) Use the Service/Capability STLSC matrix to identify the Service Thread that encompasses the system being replaced. If more than one Service Thread is supported by the system, use the Service Thread with the highest STLSC value (e.g. the Host central computer complex

supports both the CRAD surveillance Service Thread and the CFAD flight data processing Service Thread).

2) Use the availability associated with the highest SLTSC value to determine the appropriate system availability requirement. The STLSC availability value was designed to incorporate a built-in margin sufficient to ensure applicability to any component in the thread. (See discussion in Paragraph 6.5.) There is no need to perform a mathematical allocation of the end-to-end thread availability to each component in the thread, and doing so would result in an excessive system availability requirement.

3) Use the NAS-SR-1000 requirements presented in TABLE 6-5 to determine appropriate base-line MTBF, MTTR, and recovery time (if applicable) values for each of the Service Threads to ensure consistency with STLSC availabilities.

For systems that are not simple replacements of systems contained in existing Service Threads, define a new Service Thread. The appropriate STLSC Matrix for the domain and the Service Thread Reliability, Maintainability, and Recovery Times Table, TABLE 6-5, need to be updated, and a new Service Thread Diagram needs to be created and included in Appendix E.

As discussed in the preceding section, the rounding up of the availability requirements for the Service Threads provides a sufficient margin to eliminate the need for a mathematical allocation of the Service Thread availability across each component of the Service Thread; the availability requirement associated with a Service Thread also can be used as the availability requirement for a component of the thread. The practical purpose of the availability requirement is to determine fundamental system architecture issues such as whether or not fault tolerance and automatic recovery are required, and to ensure that adequate levels of redundancy will be incorporated into the system architecture. The primary driver of the actual operational availability will be the reliability of the software and automatic recovery mechanisms.

Appendix B provides charts and tables that can be used to determine the availability and reliability of repairable redundant systems, based on the availability and reliability of the redundant elements.

7.1.1.2 Remote/Distributed Subsystems

This category includes systems with Remote/Distributed elements, such as radar sites, air-to-ground communications sites and navigation aids. These systems are characterized by their spatial diversity. The surveillance and communications resources for a major facility such as a TRACON or ARTCC are provided by a number of remote sites. Failure of a remote site may or may not degrade the overall surveillance, communications, or navigation function, depending on the degree overlapping coverage, but the service and space diversity of these remote systems makes total failure virtually impossible.

Attempts have been made in the past to perform a top-down allocation to a subsystem of distributed elements. To do so requires that a hypothetical failure definition for the subsystem be defined. For example, the surveillance subsystem could be considered to be down if two out of fifty radar sites are inoperable. This failure definition is admittedly arbitrary and ignores the unique characteristics of each installation, including air route structure, geography, overlapping coverage, etc. Because such schemes rely almost entirely on “r out of n” criteria for subsystem failure definitions, the availability allocated to an individual element of a Remote/Distributed subsystem may be much lower than that which could be reasonably expected from a quality piece of equipment.

For these reasons, a top down allocation from NAS requirements to elements comprising a distributed subsystem is not appropriate, and this category of systems has been isolated as Remote/Distributed Service Threads in the STLSC matrices in FIGURE 6-7, FIGURE 6-8, and FIGURE 6-9. The RMA requirements for the individual elements comprising a Remote/Distributed subsystem should be

determined by life-cycle cost considerations and the experience of FAA acquisition specialists in dealing with realistic and achievable requirements. The overall reliability characteristics of the entire distributed subsystem are achieved through the use of diversity.

FAA Order 6000.36, "*Communication Diversity*," for example, established the national guidance to reduce the vulnerability of these Remote/Distributed services to single points of failure. The order provides for the establishment of regional Communications Working Groups (CWGs) to develop regional communications diversity plans for all pacer airports, other level 5 air traffic control facilities, and the Flight Service Data Processing System (FSDPS) services to Automated Flight Service Stations (AFSS).

The scope of FAA Order 6000.36 includes not only communications services, but also surveillance services. The NAPRS services to which the order applies are listed in Appendix 1 of the order. They correspond to the FAA Order 6040.15D services in that were mapped to the Remote/Distributed category and designated as supporting critical NAS Architecture Capabilities in the matrices in FIGURE 6-7, FIGURE 6-8, and FIGURE 6-9.

FAA Order 6000.36 defines five different diversity approaches that may be employed:

- Service Diversity – services provided via alternate sites; e.g. overlapping radar or communications coverage)
- Circuit Diversity – physical separation of cable systems by a minimum of 25 feet
- Space Diversity – antennas at different locations
- Media Diversity – radio/microwave, public telephone network, satellite, etc.
- Frequency Diversity

The type(s) and extent of diversity to be used are to be determined, based on local and regional conditions, in a bottom-up fashion by communications working groups.

FAA Order 6000.36 tends to support the approach recommended in this handbook – exempting Remote/Distributed services and systems from top-down allocation of NAS-SR-1000 availability requirements. The number and placement of the elements should be determined by FAA specialists knowledgeable in the operational characteristics and requirements for a specific facility instead of by a mechanical mathematical allocation process. Ensuring that the NAS-Level availability requirements are not degraded by failures of Remote/Distributed subsystems in a Service Thread can best be achieved through the judicious use of diversity techniques tailored to the local characteristics of a facility.

The key point in the approach for Remote/Distributed systems is that the path to achieving NAS-Level availability requirements employs diversity techniques, establishes that the RMA specifications for individual Remote/Distributed elements are an outgrowth of a business decision by FAA Service Unit, and that these decisions are based on trade-off analyses that involve factors such as what is available, what may be achievable, and how increasing reliability requirements might save on the costs of equipment operation and maintenance.

Distributed display consoles have been included in this category, since the same allocation rationale has been applied to them. For the same reasons given for remote systems, the reliability requirements for individual display consoles should be primarily a business decision determined by life cycle cost tradeoff analyses. The number and placement of consoles should be determined by operational considerations.

Airport surveillance radars are also included in this category. Even though they are not distributed like the en route radar sensors, their RMA requirements still should be determined by life cycle cost tradeoff analyses. Some locations may require more than one radar – based on the level of operations, geography and traffic patterns – but, as with subsystems with distributed elements, the decision can best be made by personnel knowledgeable in the unique operational characteristics of a given facility.

Navigation systems are remote from the air traffic control facilities and may or may not be distributed. The VOR navigation system consists of many distributed elements, but an airport instrument landing system (ILS) does not. Because the Service Threads are the responsibility of Air Traffic personnel, NAVAIDS that provide services to aircraft (and not to Air Traffic personnel) are not included in the NAPRS 6040.15 Service Threads. Again, RMA requirements for navigation systems should be determined by life-cycle cost tradeoff analyses, and the redundancy, overlapping coverage, and placement should be determined on a case-by-case basis by operational considerations determined by knowledgeable experts.

7.1.1.3 Support Systems

Support systems fall into two major categories: Infrastructure systems such as power systems, heating ventilation and air conditioning (HVAC) systems, and Mission Support systems that provide administrative support to assist the management of the NAS airspace and infrastructure. RMA requirements for these two categories are treated differently.

7.1.1.3.1 Infrastructure Systems (Power Systems)

(Currently, this Handbook addresses only the RMA requirements for power systems. Power systems are the most critical infrastructure system because not only do all of the Service Threads depend on the availability of the power system, but the HVAC and other infrastructure systems as well depend on the power system availability.)

Infrastructure systems include both power systems and heating ventilation and air conditioning (HVAC) systems. The complex interactions of infrastructure systems with the systems they support violate the independence assumption that is the basis of conventional RMA allocation and prediction. By their very nature, systems in an air traffic control facility depend on the supporting infrastructure systems for their continued operation. Failures of infrastructure systems can be a direct cause of failures in the systems they support.

Moreover, failures of infrastructure services may or may not cause failures in the Service Threads they support, and the duration of a failure in the infrastructure service is not necessarily the same as the duration of the power related failure in a supported Service Thread. A short power interruption of less than a second, for example, can cause a failure in a computer system that may disrupt operations for hours. In contrast, an interruption in HVAC service may have no effect at all on the supported services, provided that HVAC service is restored before environmental conditions deteriorate beyond what can be tolerated by the systems they support.

Because of the complex interaction of the infrastructure systems with the Service Threads they support, top-down allocations of NAS-SR-1000 availability requirements are limited to simple inherent availability requirements that can be used to determine the structure of the power system architecture. The allocated power system requirements are shown in . The inherent availability requirement for power systems at larger facilities is derived from the NAS-SR-1000 requirement of .99999 for critical capabilities. It should be emphasized that these inherent availability requirements serve only to drive the power system architectures, and should not be considered to be representative of the predicted operational availability of the power system or the Service Threads it supports.

At smaller terminal facilities, the inherent availability requirements for the Critical Power Distribution System can be derated because the reduced traffic levels at these facilities allow manual procedures to be used to compensate for power interruptions without causing serious disruptions in either safety or efficiency of traffic movement.

The smallest terminal facilities do not require a Critical Power Distribution System. The power systems at these facilities generally consist of commercial power with an engine generator or battery backup. The availability of these power systems is determined by the availability of the commercial power system components employed. Allocated NAS-SR-1000 requirements are not applicable to these systems.

The FAA Power Distribution Systems are developed using standard commercial off-the-shelf power system components whose RMA characteristics cannot be specified by the FAA. The RMA characteristics of commercial power system components are documented in IEEE Std 493-1997, *Recommended Practice for the Design of Reliable Industrial and Commercial Power Systems,* (Gold Book). This document presents the fundamentals of reliability analysis applied to the planning and design of electric power distribution systems, and contains a catalog of commercially available power system components and operational reliability data for the components. Engineers use the Gold Book and the components discussed in it to determine the configuration and architecture of power systems required to support a given level of availability. Since the RMA characteristics of the power system components are fixed, the only way power system availability can be increased is through the application of redundancy and diversity in the power system architecture.

Note: although the inherent reliability and availability of a power distribution system can be predicted to show that the power system is compliant with the allocated NAS-SR-1000 availability requirements, the dependent relationship between power systems and the systems they support precludes the use of conventional RMA modeling techniques to predict the operational reliability and availability of the power system and the Service Threads it supports.[13]

The FAA has developed a set of standard power system architectures and used computer simulation models to verify that the standard architectures comply with the derived NAS-SR-1000 requirements. The standards and operating practices for power systems are documented in FAA Order 6950.2D, *Electrical Power Policy Implementation at National Airspace System Facilities.* Since the verification of the power system architecture availability with the NAS-SR-1000 availability requirements has been demonstrated, there is no need for additional modeling efforts. All that is required is to select the appropriate architecture.

The focus on FAA power systems is on the sustainment of the existing aging power systems, many of whose components are approaching or have exceeded end-of-life expectations. There is no plan or need to redesign FAA power distribution system architectures. Therefore, the primary objectives of this Handbook with respect to power systems are to:

- Document the relationship between Service Threads and the power system architectures in FAA Order 6950.2D.
- Demonstrate that the inherent availability of existing power system architectures is consistent with the derived NAS-SR-1000 availability requirements.
- Identify potential “red flags” for terminal facilities that may be operating with inadequate power distribution systems as a consequence of traffic growth.
- Provide power system requirements for new facilities.

The matrices in FIGURE 6-7, FIGURE 6-8, and FIGURE 6-9 encapsulate the information required to achieve these objectives. It is only necessary to look at the power system architecture row(s) in the appropriate matrix to determine the required power system architecture for a facility. This will determine if the facility has the required power system architecture.

7.1.1.3.2 Mission Support Systems

Mission Support services used for airspace design and management of the NAS infrastructure are not generally real-time services and are not reportable services within NAPRS. For these reasons, it is not appropriate to allocate NAS-SR-1000 availabilities associated with real-time services used to perform the air traffic control mission to this category of services and systems.

The RMA requirements for the systems and subsystems comprising Mission Support Service Threads are determined by Acquisition Managers in accordance what is achievable and Life Cycle Cost considerations.

7.1.2 Analyzing Scheduled Downtime Requirements

During this step of the acquisition planning process, the issue of scheduled downtime for the system must be addressed. Although scheduled downtime is not included in the availability requirement discussed in the preceding paragraph, it is still an important factor in ensuring the operational suitability of the system being acquired.

The anticipated frequency and duration of scheduled system downtime to perform preventive maintenance tasks, software upgrades, and problem fixes, adaptation data changes, etc. must be considered with respect to the anticipated operational profile for the system. Some, if not most, of the scheduled downtime requirements are beyond the control of the contractor. One exception is the preventive maintenance requirements of the system hardware, including cleaning, changing filters, real-time performance monitoring, and running off-line diagnostics to detect deteriorating components.

Many NAS systems are not needed on a 24/7 basis, some airports restrict late night operations, and some weather systems are only needed during periods of adverse weather. If projected downtime requirements can be accommodated without unduly disrupting Air Traffic Control operations by scheduling downtime during low traffic periods or when the system is not needed, then there is no impact. A requirement to limit the frequency and duration of required preventive maintenance could be added to the maintainability section of the SLS. However, since most of the automation system hardware is COTS, the preventive maintenance requirements should be known in advance and will not be affected by any requirements added to the SLS. Therefore, additional SLS maintainability requirements are only appropriate for custom-developed hardware.

Conversely, if scheduled downtime cannot be accommodated without disrupting air traffic control operations, it is necessary to re-examine the approach being considered. It also may be necessary to add an independent backup system to supply the needed service while the primary system is unavailable.

7.1.3 Modifications to STLSC Levels

It may be desirable or appropriate to acquire or implement a system that has an availability value different from its associated SLTSC availability level. In such instances, STLSC availability values can be exceeded but not reduced. That is, the desired system must at least meet the STLSC “floor value” availability requirements for the associated Service Thread. While availability requirements for the system can be defined higher than the SLTSC rating, the acquisition team must accept the potential cost

implications of raising the availability level. It is *not appropriate*, however, to lower the availability level of a system below that required by the thread's SLTSC.

7.1.4 Redundancy and Fault Tolerance Requirements

The first determinant of the need for redundancy and fault tolerance is the required inherent availability of the hardware architecture. If the failure and repair rates of a single set of system elements cannot meet the inherent availability requirements, redundancy and automatic fault detection and recovery mechanisms must be added. There must be an adequate number of hardware elements that, given their failure and repair rates, the combinatorial probability of running out of spares is consistent with the inherent availability requirements.

There are other reasons beyond the inherent availability of the hardware architecture that may dictate a need for redundancy and/or fault tolerance. Even if the system hardware can meet the inherent hardware availability, redundancy may be required to achieve the required recovery times and provide the capability to recover from software failures.

All Service Threads with a STLSC of "Efficiency-Critical" have rapid recovery time requirements because of the potentially severe consequences of lengthy service interruptions on the efficiency of NAS operations. These recovery time requirements will, in all probability, call for the use of redundancy and fault- tolerant techniques. The lengthy times associated with rebooting a computer to recover from software failures or "hangs" indicates a need for a standby computer that can rapidly take over from a failed computer.

7.1.5 Preliminary Requirements Analysis Checklist

- ☐ Determine the category of the system being acquired from the Taxonomy Chart.
- ☐ For Information Systems, identify the Service Thread containing the system to be acquired.
- ☐ Determine inherent availability requirements from the NAS-SR-1000 matrix corresponding to **Error! Reference source not found.**that Service Thread.
- ☐ Determine the RMA requirements for that Service Thread from the table in the NAS-SR-1000 corresponding to TABLE 6-5 in this handbook.
- ☐ For power systems, determine the availability requirements according to the highest STLSC of the Service Threads being supported and the Facility Level from the table in the NAS-SR-1000.corresponding to in this handbook.
- ☐ Select a standard power system configuration that will meet the availability requirements.
- ☐ For remote communications links use the requirements in Section 5.4.4 of the NAS-SR-1000.
- ☐ The RMA requirements for other distributed subsystems such as radars, air to ground communications, and display consoles are not derived from NAS-Level NAS-SR-1000 requirements. They are determined by technical feasibility and life cycle cost considerations.

7.2 Procurement Package Preparation

The primary objectives to be achieved in preparing the procurement package are as follows:

- To provide the specifications that define the RMA and fault tolerance requirements for the delivered system and form the basis of a binding contract between the successful offeror and the Government.
- To define the effort required of the contractor to provide the documentation, engineering, and testing required to monitor the design and development effort, and to support risk management, design validation, and the testing of reliability growth activities.
- To provide guidance to prospective offerors concerning the content of the RMA sections of the technical proposal, including design descriptions and program management data required to facilitate the technical evaluation of the offerors' fault-tolerant design approach, risk management, software fault avoidance and reliability growth programs.

7.2.1 System-Level Specification

The System-Level specification serves as the contractual basis for defining the design characteristics and performance that are expected of the system. From the standpoint of fault tolerance and RMA characteristics, it is necessary to define the quantitative RMA and performance characteristics of the automatic fault detection and recovery mechanisms. It is also necessary to define the operational requirements needed to permit FAA facilities personnel to perform real-time monitoring and control and manual recovery operations as well as diagnostic and support activities.

While it is not appropriate to dictate specifics as to the system design, it is important to take operational needs and system realities into account. These characteristics are driven by operational considerations of the system and could affect its ability to participate in a redundant relationship with another Service Thread. Examples include limited numbers of consoles and limitations on particular consoles to accomplish particular system functions.

A typical System-Level Specification prepared in accordance with MIL-STD-961E will contain the following sections:

1. Scope
2. Applicable Documents
3. Requirements
4. Verification (or Qualification)
5. Packaging
6. Notes

The sections relevant to RMA are Section 3, "Requirements," and Section 4, "Verification (or Qualification). The organization of subsections within these sections can vary, but generally, RMA requirements appear in three general categories within Section 3:

1. System Quality Factors
2. System Design Characteristics

3. System Operations

Automation systems also include a separate subsection on the functional requirements for the computer software. Functional requirements may include RMA-related requirements for monitoring and controlling system operations. Each of these sections will be presented separately. This section and Appendix A contains checklists and/or sample requirements. These forms of guidance are presented for use in constructing a tailored set of SLS requirements. The reader is cautioned not to use the requirements verbatim, but instead to use them as a basis for creating a system-specific set of SLS requirements.

7.2.1.1 System Quality Factors

System Quality Factors contain quantitative requirements specifying characteristics such as reliability, maintainability, and availability, as well performance requirements for data throughput and response times.

Availability

The availability requirements to be included in the SLS are determined by the procedures described in Section 7.1.

The availability requirements in the SLS are built upon inherent hardware availability. The inherent availability represents the theoretical maximum availability that could be achieved by the system if automatic recovery were one hundred percent effective and there were no failures caused by latent software defects. This construct strictly represents the theoretical availability of the system hardware based only on the reliability (MTBF) and maintainability (MTTR) of the hardware components and the level of redundancy provided. It does not include the effects of scheduled downtime, shortages of spares, or unavailable or poorly trained service personnel.

Imposing an inherent availability requirement only serves to ensure that the proposed hardware configuration is *potentially* capable of meeting the NAS-Level requirement, based on the reliability and maintainability characteristics of the system components and the redundancy provided. Inherent availability is not a testable requirement. Verification of compliance with the inherent availability requirement is substantiated by the use of straightforward combinatorial availability models that are easily understood by both contractor and government personnel. The contractor must, of course, supply supporting documentation that verifies the realism of the component or subsystem MTBF and MTTR values used in the model.

The inherent availability of a single element is based on the following equation:

$$A = \frac{MTBF}{MTBF + MTTR} \qquad [7\text{-}1]$$

The inherent availability of a string of elements, all of which must be up for the system to be up, is given by:

$$A_T = A_1 A_2 A_3 \cdots A_n \qquad [7\text{-}2]$$

The inherent availability of a two-element redundant system (considered operational if both elements are up, or if the first is up and the second is down, or if the first is down and the second is up) is given by:

$$A_{Inherent} = (A_1 A_2 + A_1 \overline{A}_2 + \overline{A}_1 A_2) \qquad [7\text{-}3]$$

$$A_{Inherent} = (1 - \overline{A}_1 \overline{A}_2) \quad [7\text{-}4]$$

(Where $\overline{A} = (1 - A)$ or the probability that an element is not available.)

The above equations are straightforward, easily understood, and combinable to model more complex architectures. They illustrate that the overriding goal for the verification of compliance with the inherent availability requirement should be to "keep it simple." Since this requirement is not a significant factor in the achieved operational reliability and availability of the delivered system, the effort devoted to it need not be more than a simple combinatorial model as in Equation [7-4], or a comparison with the tabulated values in Appendix B. This is simply a necessary first step in assessing the adequacy of a proposed hardware architecture. Attempting to use more sophisticated models to "prove" compliance with operational requirements is misleading, wastes resources, and diverts attention from addressing more significant problems that can significantly impact the operational performance of the system.

The inherent availability requirement provides a common framework for evaluating repairable redundant system architectures. In a System-Level Specification, this requirement is intended to ensure that the theoretical availability of the hardware architecture can meet key operational requirements. Compliance with this requirement is verified by simple combinatorial models. The inherent availability requirement is only a preliminary first step in a comprehensive plan that is described in the subsequent sections to attempt to ensure the deployment of a system with operationally suitable RMA characteristics.

The use of the inherent availability requirement is aimed primarily at Service Threads with a STLSC level of "efficiency-critical." (As discussed in Paragraph 6.3, any Service Threads assessed as potentially "safety-critical" must be decomposed into two "efficiency-critical" Service Threads.) Systems participating in threads with an "efficiency-critical" STLSC level will likely employ redundancy and fault tolerance to achieve the required inherent availability and recovery times. The combined availability of a two element redundant configuration is given by Equation [7-4]. The use of inherent availability as a requirement for systems participating in Service Threads with a STLSC level of "essential" and not employing redundancy can be verified with the basic availability equation of Equation [7-1].

Reliability

Most of the hardware elements comprising modern automation systems are commercial off-the-shelf products. Their reliability is a "given." True COTS products are not going to be redesigned for FAA acquisitions. Attempting to do so would significantly increase costs and defeat the whole purpose of attempting to leverage commercial investment. There are, however, some high-level constraints on the element reliability that are imposed by the inherent availability requirements in the preceding paragraphs.

For hardware that is custom-developed for the FAA, it is inappropriate to attempt a top-level allocation of NAS-Level RMA requirements. Acquisition specialists who are cognizant of life cycle cost issues and the current state-of-the-art for these systems can best establish their reliability requirements.

For redundant automation systems, the predominant sources of unscheduled interruptions are latent software defects. For systems extensive newly developed software, these defects are an inescapable fact of life. For these systems, it is unrealistic to attempt to follow the standard military reliability specification and acceptance testing procedures that were developed for electronic equipment having comparatively low reliability. These procedures were developed for equipment that had MTBFs on the order of a few hundred hours. After the hardware was developed, a number of pre-production models would be locked in a room and left to operate for a fixed period of time. At the end of the period, the Government would determine the number of equipments still operating and accept or reject the design based on proven statistical decision criteria.

Although it is theoretically possible to insert any arbitrarily high reliability requirement into a specification, it should be recognized that the resulting contract provision would be unenforceable. There are several reasons for this. There is a fundamental statistical limitation for reliability acceptance tests that is imposed by the number of test hours needed to obtain a statistically valid result. A general “rule of thumb” for formal reliability acceptance tests is that the total number of test hours should be about ten times the required MTBF. As the total number of hours available for reliability testing is reduced below this value, the range of uncertainty about the value of true MTBF increases rapidly, as does the risk of making an incorrect decision about whether or not to accept the system. (The quantitative statistical basis for these statements is presented in more detail in Appendix C.)

For “efficiency-critical” systems, the required test period for one system could last hundreds of years. Alternatively, a hundred systems could be tested for one year. Neither alternative is practical. The fact that most of the failures result from correctable software mistakes that should not reoccur once they are corrected also makes a simple reliability acceptance test impractical. Finally, since it is not realistic to terminate the program based on the result of a reliability acceptance test, the nature and large investment of resources in major system acquisitions makes reliability compliance testing impractical.

In the real world, the only viable option is to keep testing the system and correcting problems until the system becomes stable enough to send to the field – or the cost and schedule overruns cause the program to be restructured or terminated. To facilitate this process a System-Level driver, with repeatable complex ATC scenarios, is valuable. In addition, a data extraction and data reduction and analysis (DR&A) process that assists in ferreting out and characterizing the latent defects is also necessary.

It would be wrong to conclude there should be no reliability requirements in the SLS. Certainly, the Government needs reliability requirements to obtain leverage over the contractor and ensure that adequate resources are applied to expose and correct latent software defects until the system reaches an acceptable level of operational reliability. Reliability growth requirements should be established that define the minimum level of reliability to be achieved before the system is deployed to the first site, and a final level of reliability that must be achieved by the final site. The primary purpose of these requirements is to serve as a metric that indicates how aggressive the contractor has been at fixing problems as they occur. The FAA customarily documents problems observed during testing as Program Trouble Reports (PTRs).

TABLE 6-5 provides an example of the NAS-SR-1000 requirements for the MTBF, MTTR, and recovery time for each of the Service Threads.

For systems employing automatic fault detection and recovery, the reliability requirements are coupled to the restoration time. For example, if a system is designed to recover automatically within t seconds, there needs to be a limit on the number of successful automatic recoveries, i.e. an MTBF requirement for interruptions that are equal to, or less than, t seconds. A different MTBF requirement is established for restorations that take longer than t seconds, to address failures for which automatic recovery is unsuccessful.

The establishment of the MTBF and recovery time requirements in TABLE 6-5 draws upon a synthesis of operational needs, the measured performance of existing systems, and the practical realities of the current state of the art for automatic recovery. The reliability requirements, when combined with a 30 minute MTTR using Equation [7-1] yields availabilities that meet or exceed the inherent availability requirements for the Service Threads.

The allowable recovery times were developed to balance operational needs with practical realities. While it is operationally desirable to make the automatic recovery time as short as possible, reducing the recovery time allocation excessively can impose severe restrictions on the design and stability of the fault

tolerance mechanisms. It also can dramatically increase the performance overhead generated by the steady state operation of error detecting "heartbeats" and other status monitoring activities.

Although some automatic recoveries can be completed quickly, recoveries that require a complete system "warm start," or a total system reboot, can take much longer. These recovery times are determined by factors such as the size of the applications and operating system, and the speed of the processor and associated storage devices. There are only a limited number of things that can be done to speed up recovery times that are driven by hardware speed and the size of the program.

The reliability MTBF requirements can be further subdivided to segregate failures requiring only a simple application restart or system reconfiguration from those that require a warm start or a complete system reboot.

The MTBF requirements are predicated on the assumption that any system going to the field should be at least as good as the system it replaces. Target requirements are set to equal the reliability of currently fielded systems, as presented in the 6040.20 NAPRS reports.

The MTBF values in the table represent the final steady-state values at the end of the reliability growth program, when the system reaches operational readiness. However, it is both necessary and desirable to begin deliveries to the field before this final value is reached. The positive benefits of doing this are that testing many systems concurrently increases the overall number test hours, and field testing provides a more realistic test environment. Both of these factors tend to increase the rate of exposure of latent software defects, accelerate the reliability growth rate, and build confidence in the system's reliability.

The NAS-SR-1000 reliability values in TABLE 6-5 refer to STLSC specifically associated with the overall Service Threads, but because of the margins incorporated in the Service Thread availability allocation, the reliability values (MTBFs) in TABLE 6-5 can be applied directly to any system in the thread. When incorporating the NAS-SR-1000 reliability values into a SLS, these should be the final values defined by some program milestone, such as delivery to the last operational site, to signal the end of the reliability growth program. To implement a reliability growth program, it is necessary to define a second set of MTBF requirements that represent the criteria for beginning deliveries to operational sites. The values chosen should represent a minimum level of system stability acceptable to field personnel. FAA field personnel need to be involved both in establishing these requirements and in their testing at the WJHTC. Involvement of field personnel in the test process will help to build their confidence, ensure their cooperation, and foster their acceptance of the system.

Appendix A provides examples of reliability specifications that have been used in previous procurements. They may or may not be appropriate for any given acquisition. They are intended to be helpful in specification preparation.

Maintainability

Maintainability requirements traditionally pertain to such inherent characteristics of the hardware design as the ability to isolate, access, and replace a failed component. These characteristics are generally fixed for COTS components. The inherent availability requirements in Paragraph 6.5 impose some constraints on maintainability because the inherent availability depends on the hardware MTBF and MTTR and the number of redundant elements. In systems constructed with COTS hardware, the MTTR is considered to be the time required to remove and replace all or a spared element of the COTS hardware. Additional maintainability requirements may be specified in this section provided they do not conflict with the goal to employ COTS hardware whenever practical.

The FAA generally requires a Mean Time to Repair of 30 minutes. For systems using COTS hardware, the MTTR refers to the time required to remove and replace the COTS hardware

System Performance Requirements

System performance and response times are closely coupled to reliability issues. The requirement to have rapid and consistent automatic fault detection and recovery times imposes inflexible response time requirements on the internal messages used to monitor the system's health and initiate automatic recovery actions. If the allocated response times are exceeded, false alarms may be generated and inconsistent and incomplete recovery actions will result.

At the same time, the steady state operation of the system monitoring and fault tolerance heartbeats imposes a significant overhead on the system workload. The system must be designed with sufficient reserve capacity to be able to accommodate temporary overloads in the external workload or the large numbers of error messages that may result during failure and recovery operations. The reserve capacity also must be large enough to accommodate the seemingly inevitable software growth and overly optimistic performance predictions and model assumptions.

Specification of the automatic recovery time requirements must follow a synthesis of operational needs and the practical realities of the current performance of computer hardware. There is a significant challenge in attempting to meet stringent air traffic control operational requirements with imperfect software running on commercial computing platforms. The FAA strategy has been to employ software fault tolerance mechanisms to mask hardware and software failures.

A fundamental tradeoff must be made between operational needs and performance constraints imposed by the hardware platform. From an operational viewpoint, the recovery time should be as short as possible, but reducing the recovery time significantly increases the steady state system load and imposes severe constraints on the internal fault tolerance response times needed to ensure stable operation of the system.

Although it is the contractor's responsibility to allocate recovery time requirements to lower level system design parameters, attempting to design to unrealistic parameters can significantly increase program risk. Ultimately, it is likely that the recovery time requirement will need to be reduced to an achievable value. It is preferable to avoid the unnecessary cost and schedule expenses that result from attempting to meet an unrealistic requirement. While the Government always should attempt to write realistic requirements, it also must monitor the development effort closely to continually assess the contractor's performance and the realism of the requirement. A strategy for accomplishing this is presented in Paragraph 7.4.3.3.

Once the automatic recovery mechanisms are designed to operate within a specific recovery time, management must recognize that there are some categories of service interruptions that cannot be restored within the specified automatic recovery time. The most obvious class of this type of failure is a hardware failure that occurs when a redundant element is unavailable. Other examples are software failures that cause the system to hang, unsuccessful recovery attempts, etc. When conventional recovery attempts fail, it may be necessary to reboot some computers in the system and may or may not require specialist intervention.

The recommended strategy for specifying reliability requirements that accommodate these different categories of failures is to establish a separate set of requirements for each failure category. Each set of requirements should specify the duration of the interruption and the allowable MTBF for a particular type of interruption. For example:

- Interruptions that are recovered automatically within the required recovery time
- Interruptions that require reloading software
- Interruptions that require hardware repair or replacement

7.2.1.2 System Design Characteristics

This section of the SLS contains requirements related to design characteristics of hardware and software that can affect system reliability and maintainability. Many of these requirements will be unique to the particular system being acquired.

7.2.1.3 System Operations

This section of the SLS contains RMA-related requirements for the following topics:

- Monitor and Control (M&C) - The Monitor and Control function is dual purpose. It contains functionality to automatically monitor and control system operation, and it contains functionality that allows a properly qualified specialist to interact with the system to perform monitor and control system operations, system configuration, system diagnosis and other RMA related activities. Design characteristics include functional requirements and requirements for the Computer/Human Interface (CHI) with the system operator.

- System Analysis Recording (SAR) - The System Analysis and Recording function provides the ability to monitor system operation, record the monitored data, and play it back at a later time for analysis. SAR data is used for incident and accident analysis, performance monitoring and problem diagnosis.

- Startup/Startover - Startup/Startover is one of the most critical system functions and has a significant impact on the ability of the system to meet its RMA requirements, especially for software intensive systems.
- Software Deployment, Downloading, and Cutover - Software Loading and Cutover is a set of functions associated with the transfer, loading and cutover of software to the system. Cutover could be to a new release or a prior release.

- Certification - Certification is an inherently human process of analyzing available data to determine if the system is worthy of performing its intended function. One element of data is often the results of a certification function that is designed to exercise end-to-end system functionality using known data and predicable results. Successful completion of the certification function is one element of data used by the Specialist to determine the system is worthy of certification. Some systems employ a background diagnostic or verification process to provide evidence of continued system certifiability.
- Transition – Transition is a set of requirements associated with providing functionality required to support the transition to upgraded or new systems.

- Maintenance Support – Maintenance support is a collection of requirements associated with performing preventative and corrective maintenance of equipment and software.

- Test Support – Test support is a collection of requirements associated with supporting system testing before, during and after installation of the system. System-Level drivers capable of simulating realistic and stressful operations in a test environment and a data extraction and analysis capability for recording and analyzing test data are both essential components in an aggressive reliability growth program. Requirements for additional test support tools that are not in System Analysis Recording should be included here.

- M&C Training – Training support is a collection of requirements associated with supporting training of system specialists.

7.2.1.4 System-Level Specification RMA Checklist

- ☐ Include NAS-SR-1000 inherent availability requirements.
- ☐ Include NAS-SR-1000 MTBF, MTTR, and recovery time requirements.
- ☐ Develop initial MTBF criteria for shipment of the system to the first operational site.
- ☐ Consider potential need for additional RMA quality factors for areas such as Operational Positions, Monitor & Control Positions, Data Recording, Operational Transition, etc.
- ☐ Review checklists of potential design characteristics.
- ☐ Review checklists of potential requirements for System Operations.
- ☐ Incorporate requirements for test tools such as System-Level Drivers and Data Extraction and Analysis to support a reliability growth program.
- ☐ Ensure the RMA requirements for other distributed subsystems such as radars, air to ground communications, and display consoles are not derived from NAS-Level NAS-SR-1000 requirements. These requirements must be determined by technical feasibility and life cycle cost considerations.

7.2.2 Statement of Work

The Statement of Work describes the RMA-related tasks required of the contractor to design, analyze, monitor risk, implement fault avoidance programs, and prepare the documentation and engineering support required to provide Government oversight of the RMA, Monitor and Control function, fault tolerant design effort, support fault tolerance risk management and conduct reliability growth testing. Typical activities to be called out include:

- Conduct Technical Interchange Meetings (TIMs)
- Prepare Documentation and Reports, e.g.,
  - RMA Program Plans
  - RMA Modeling and Prediction Reports
  - Failure Modes and Effects Analysis
- Perform Risk Reduction Activities
- Develop Reliability Models
- Conduct Performance Modeling Activities
- Develop a Monitor and Control Design

7.2.2.1 Technical Interchange Meetings

The following text is an example of an SOW requirement for technical interchange meetings:

> *The Contractor shall conduct and administratively support periodic Technical Interchange Meetings (TIMs) when directed by the Contracting Officer. TIMs may also*

be scheduled in Washington, DC, Atlantic City, NJ, or at another location approved by the FAA. TIMs may be held individually or as part of scheduled Program Management Reviews (PMRs). During the TIMs the Contractor and the FAA will discuss specific technical activities, including studies, test plans, test results, design issues, technical decisions, logistics, and implementation concerns to ensure continuing FAA visibility into the technical progress of the contract. The Contractor shall document TIMs in CDRL A-xxx.

This generic SOW language may be adequate to support fault tolerance TIMs, without specifically identifying the fault tolerance requirements. The need for more specific language should be discussed with the Contracting Officer.

7.2.2.2 Documentation

The documentation required to support RMA and Fault Tolerance design monitoring includes formal documentation such as RMA program plans, RMA modeling and prediction reports and other standardized reports for which the FAA has standard Data Item Descriptions (DIDs). Table XXXX depicts typical DIDs, their Title, Description and Application.

TABLE 7-1: RMA-Related Data Item Descriptions

DID Ref. No.	Title	Description	Applicability/Interrelationship	Relevance to RMA
(B008)	System Performance Plan	The purpose of the System Performance Plan is to document: (1) the performance-related Technical Performance Measures (TPMs) to be tracked; (2) the models, prototypes, or other techniques the Contractor proposes to use to determine TPM values; (3) how the models, prototypes, or other techniques will be validated and verified; (4) the plan for collecting performance data during system development; and (5) the interactions among engineering groups and software developers that must occur to implement the system performance plan.	The System Performance Plan and the System Performance Reports document the system performance engineering process. The System Performance Plan describes strategy for predicting TPM values. The System Performance Reports document the actual results from applying the predictive models and identify performance risk mitigation strategies, where required.	This DID should be reviewed by RMA personnel. The report shall identify the set of Technical Performance Measures (TPMs) to be tracked. The report shall demonstrate that the proposed set of TPMs is sufficient for effective performance risk management. The report shall identify the techniques that will be used to estimate the values of the TPMs during system development.
(B009)	System Performance Report	System Performance Reports document: (1) currently estimated values of Technical Performance Measures (TPMs); (2) TPM Variance Analysis Reports (TVARs) and corresponding Risk Mitigation Plans; (3) uncertainties or deficiencies in TPM estimates; and (4) allocation of performance requirements to hardware and software elements that result in an operational system capable of meeting all performance requirements while processing the Design Workload. The System Performance Reports provide early insight into the system's ability to meet specified performance requirements.	The System Performance Plan and the System Performance Reports document the system performance engineering process. The System Performance Plan describes the strategy for predicting TPM values. The System Performance Reports document actual results from applying the strategy.	This DID should be reviewed by RMA personnel. See (B008) above.
(B021)	Reliability Maintainability Availability (RMA) Modeling and Prediction Report	The purpose of the Reliability Maintainability Availability (RMA) Modeling and Prediction Report is to document analysis results and supporting assumptions that demonstrate that the Contractor's proposed system design will satisfy the Reliability, Maintainability, and Availability (RMA) requirements in the System Specification Document (SSD).	The models and predictions documented in this report are used to assess system compliance with the RMA requirements contained in the SSD, identify areas of risk, support generation of Maintenance Plans, and support logistics planning and cost studies. The models and analyses documented in this report also support the reliability growth projections contained in the Failure Data and Corrective Action Summary Report. The combination of the	This review of this DID should be the responsibility of RMA personnel. The report shall document the results of analysis of the proposed system's ability to satisfy the reliability design requirements of the SSD. The report shall document the results of analysis of the proposed system's ability to satisfy the maintainability design requirements of the SSD. The report shall document the results of

These reports must be generated according to a delivery schedule that is part of the contract. The timing and frequency of these reports should be negotiated to match the progress of the development of the fault-tolerant design. The fact that these CDRL items are contract deliverables, upon which contractual performance is measured, limits their usefulness.

What is more useful for the purposes of monitoring the progress of the fault-tolerant design is informal documentation that is used for internal communication between members of the contractor's design team. Acquisition managers should develop strategies for minimizing formal "boilerplate" CDRL items and devise strategies for obtaining Government access to real-time documentation of the evolving design.

7.2.2.3 Risk Reduction Activities

The SOW must include adequate levels of contractor support for measurement and tracking of critical fault tolerance design parameters and risk reduction demonstrations. These activities are further described in Section 7.4.3.

7.2.2.4 Reliability Modeling

Reliability modeling requirements imposed on the contractor should be limited to simple combinatorial availability models that demonstrate compliance with the inherent availability requirement. Complex models intended to predict the reliability of undeveloped software and the effectiveness of fault tolerance mechanisms are highly sensitive to unsubstantiated assumptions, tend to waste program resources, and generate a false sense of complacency.

7.2.2.5 Performance Modeling

In contrast to reliability modeling, performance modeling can be a valuable tool for monitoring the progress of the design. The success of the design of the fault tolerance mechanisms is highly dependent on the response times for internal health and error messages. The operation of the fault tolerance mechanisms in turn can generate a significant processing and communications overhead.

It is important that the Statement of Work include the requirement to continually maintain and update workload predictions, software processing path lengths, and processor response time and capacity predictions. Although performance experts generally assume lead on performance modeling requirements, these requirements should be reviewed to ensure that they satisfy the RMA/fault-tolerant needs.

7.2.2.6 Monitor and Control Design Requirement

The specification of the Monitor and Control requirements is a particularly difficult task, since the overall system design is either unknown at the time the specification is being prepared, or, in the case of a design competition, there are two or more different designs. In the case of competing designs, the specification must not include detail that could be used to transfer design data between offerors. The result is that the SLS requirements for the design of the M&C position are likely to be too general to be very effective in giving the Government the necessary leverage to ensure an effective user interface for the monitoring and control of the system.

The unavoidable ambiguity of the requirements is likely to lead to disagreements between the contractor and the Government over the compliance of the M&C design unless the need to jointly evolve the M&C design after contract award is anticipated and incorporated into the SOW.

(An alternative way of dealing with this dilemma is presented in Section 7.2.3.2. That is to require the offerors to present a detailed design in their proposals and incorporate the winner's design into the contractual requirements.)

7.2.2.7 Fault Avoidance Strategies

The Government may want to mandate that the contractor employ procedures designed to uncover fault tolerance design defects such as fault tree analysis or failure modes and effects analysis. However, caution should be used in mandating these techniques for software developments, as they are more generally applied to weapons systems or nuclear power plants where cause and effect are more obvious than in a decision support system.

It is assumed that more general fault avoidance strategies such as those used to promote software quality will be specified by software engineering specialists independent of the RMA/Fault Tolerance requirements.

7.2.2.8 Reliability Growth

Planning for an aggressive reliability growth program is an essential part of the development and testing of software-intensive systems used in critical applications. As discussed in Section 5, it is no longer practical to attempt a legalistic approach to enforce contractual compliance with the reliability requirements for high reliability automation systems. The test time required to obtain a statistically valid sample on which to base an accept/reject decision would be prohibitive. The inherent reliability of an automation system architecture represents potential maximum reliability if the software is perfect. The achieved reliability of an automation system is limited by undiscovered latent software defects causing system failures. The objective of the reliability growth program is to expose and correct latent software defects so that the achieved reliability approaches the inherent reliability.

The SLS contains separate MTBF values for the first site and the last site that can be used as metrics representing two points on the reliability growth curve. These MTBF values are calculated by dividing the test time by the number of failures. Because a failure review board will determine which failures are considered relevant and also expunge failures that have been fixed or that do not reoccur during a specified interval, there is a major subjective component in this measure. The MTBF obtained in this manner should not be viewed as a statistically valid estimate of the true system MTBF. If the contractor fixes the cause of each failure soon after it occurs, the MTBF could be infinite because there are no open trouble reports – even if the system is experiencing a failure every day. The MTBF calculated in this manner should be viewed as metrics that measure a contractor's responsiveness in fixing problems in a timely manner. The MTBF requirements are thus an important component in a successful reliability growth program.

The SOW needs to specify the contractor effort required to implement the reliability growth program.

The SLS needs to include requirements for the additional test tools, simulators, data recording capability, and data reduction and analysis capability that will be required to support the reliability growth program.

7.2.2.9 Statement of Work Checklist

- ☐ Provide for RMA and Fault Tolerance Technical Interchange Meetings (TIMs).
- ☐ Define CDRL Items and DIDs to provide the documentation needed to monitor the development of the fault-tolerant design and the system's RMA characteristics.
- ☐ Provide for Risk Reduction Demonstrations of critical elements of the fault-tolerant design.
- ☐ Limit required contractor RMA modeling effort to basic one-time combinatorial models of inherent reliability/availability of the system architecture.
- ☐ Incorporate requirements for continuing performance modeling to track the processing overhead and response times associated with the operation of the fault tolerance mechanisms, M&C position, and data recording capability.
- ☐ Provide for contractor effort to evolve the M&C design in response to FAA design reviews.
- ☐ Provide for contractor effort to use analytical tools to discover design defects during the development.
- ☐ Provide for contractor support for an aggressive reliability growth program.

7.2.3 Information for Proposal Preparation

The Information for Proposal Preparation (IFPP) describes material that the Government expects to be included in the offeror's proposal. The following information should be provided to assist in the technical evaluation of the fault tolerance and RMA sections of the proposal.

7.2.3.1 Inherent Availability Model

A simple inherent availability model should be included to demonstrate that the proposed architecture is compliant with the NAS-Level availability requirement. The model's input parameters include the element MTBF and MTTR values and the amount of redundancy provided. The offeror should substantiate the MTBF and MTTR values used as model inputs, preferably with field data for COTS products, or with reliability and maintainability predictions for the individual hardware elements.

7.2.3.2 Proposed M&C Design Description and Specifications

As discussed in Section 4.2.6, it will be difficult or impossible for the Government to incorporate an unambiguous specification for the M&C position into the SLS. This is likely to lead to disagreements between the contractor and the Government concerning what is considered to be compliant with the requirements.

There are two potential ways of dealing with this. One is to request that offerors propose an M&C design that is specifically tailored to the needs of their proposed system. The M&C designs would be evaluated as part of the proposal technical evaluation. The winning contractor's proposed M&C design would then be incorporated into the contract and made contractually binding.

Traditionally, the FAA has not used this approach, although it is commonly used in the Department of Defense. The approach satisfies two important objectives. It facilitates the specification of design-dependent aspects of the system and it encourages contractor innovation.

The other is to attempt to defer specification of the M&C function until after contract award, have the contractor propose an M&C design, review the approach and negotiate a change to the contract to incorporate the approved approach.

The selection of either approach should be explored with the FAA Contracting Officer.

7.2.3.3 Fault-Tolerant Design Description

The offeror's proposal should include a complete description of the proposed design approach for redundancy management and automatic fault detection and recovery. The design should be described qualitatively. In addition, the offeror should provide quantitative substantiation that the proposed design can comply with the recovery time requirements.

The offeror should also describe the strategy and process for incorporating fault tolerance mechanisms in the application software to handle unwanted, unanticipated, or erroneous inputs and responses.

7.3 Proposal Evaluation

The following topics represent the key factors in evaluating each offeror's approach to developing a system that will meet the operational needs for reliability and availability.

7.3.1 Reliability Modeling and Assessment

The evaluation of the offeror's inherent availability model is simple and straightforward. All that is required is to confirm that the model accurately represents the architecture and that the mathematical formulas are correct. The substantiation of the offeror's MTBF and MTTR values used as inputs to the model should be also reviewed and evaluated. Appendix B provides tables and charts that can be used to check each offeror's RMA model.

7.3.2 Fault-Tolerant Design Evaluation

The offeror's proposed design for automatic fault detection and recovery/redundancy management should be evaluated for its completeness and consistency. A critical factor in the evaluation is the substantiation of the design's compliance with the recovery time requirements.

There are key two aspects of the fault-tolerant design. The first is the design of the infrastructure component that contains the protocols for health monitoring, fault detection, error recovery, and redundancy management.

Equally important is the offeror's strategy for incorporating fault tolerance into the application software. Unless fault tolerance is embedded into the application software, the ability of the fault-tolerant infrastructure to effectively mask software faults will be severely limited. The ability to handle unwanted, unanticipated, or erroneous inputs and responses must be incorporated during the development of the application software.

7.3.3 Performance Modeling and Assessment

An offeror should present a complete model of the predicted system loads, capacity, and response times. Government experts in performance modeling should evaluate these models. Fault tolerance evaluators should review the models in the following areas:

Latency of fault tolerance protocols. The ability to respond within the allocated response time is critical to the success of the fault tolerance design. It should be noted that, at the proposal stage, the level of the design may not be adequate to address this issue.

System Monitoring Overhead and Response Times. The offeror should provide predictions of the additional processor loading generated to support both the system monitoring performed by the M&C function as well as by the fault tolerance heartbeat protocols and error reporting functions. Both steady-state loads and peak loads generated during fault conditions should be considered.

Relation to Overall System Capacity and Response Times. The system should be sized with sufficient reserve capacity to accommodate peaks in the external workload without causing slowdowns in the processing of fault tolerance protocols. Adequate memory should be provided to avoid paging delays that are not included in the model predictions.

7.4 Contractor Design Monitoring

7.4.1 Formal Design Reviews

Formal design reviews are a contractual requirement. Although these reviews are often too large and formal to include a meaningful dialog with the contractor, they do present an opportunity to escalate technical issues to management's attention.

7.4.2 Technical Interchange Meetings

The contractor's design progress should be reviewed in monthly Fault Tolerance TIMs. In addition to describing the design, the TIM should address the key timing parameters governing the operation of the fault tolerance protocols, the values allocated to the parameters, and the results of model predictions and or measurements made to substantiate the allocations.

7.4.3 Risk Management

The objective of the fault tolerance risk management activities is to expose flaws in the design as early as possible, so that they can be corrected "off the critical path" without affecting the overall program cost and schedule. Typically, major acquisition programs place major emphasis on formal design reviews such as the System Requirements Review (SRR), the System Design Review (SDR) the Preliminary Design Review (PDR), and the Critical Design Review (CDR). After the CDR has been successfully completed, lists of Computer Program Configuration Items (CPCIs) are released for coding, beginning the implementation phase of the contract. After CDR, there are no additional formal technical software reviews until the end of implementation phase when the Functional and Physical Configuration Audits (FCA and PCA) and formal acceptance tests are conducted.

Separate fault tolerance risk management activities should be established for:

- Fault-tolerant infrastructure

- Error handling in software applications
- Performance monitoring

The fault-tolerant infrastructure will generally be developed by individuals whose primary objective is to deliver a working infrastructure. Risk management activities associated with the infrastructure development are directed toward uncovering logic flaws and timing/performance problems.

In contrast, application developers are *not* primary concerned with fault tolerance. Their main challenge is to develop the functionality required of the application. Under schedule pressure to demonstrate the required functionality, building in the fault tolerance capabilities that need to be embedded into the application software is often overlooked or indefinitely postponed during the development of the application. Once the development has been largely completed, it can be extremely difficult to incorporate fault tolerance into the applications after the fact. Risk management for software application fault tolerance consists of establishing standards for applications developers and ensuring that the standards are followed.

Risk management of performance is typically focused on the operational functionality of the system. Special emphasis needs to be placed on the performance monitoring risk management activity to make sure that failure, failure recovery operations, system initialization/re-initialization, and switchover characteristics are properly modeled.

7.4.3.1 Fault-Tolerance Infrastructure Risk Management

The development of a fault-tolerant infrastructure primarily entails constructing mechanisms that monitor the health of the system hardware and software as well as provide the logic to switch, when necessary, to redundant elements.

The primary design driver for the fault tolerance infrastructure is the required recovery time. Timing parameters must be established to achieve a bounded recovery time, and the system performance must accommodate the overhead associated with the fault tolerance monitoring and deliver responses within established time boundaries. The timing budgets and parameters for the fault-tolerant design are derived from this requirement. The fault-tolerant timing parameters, in turn, determine the steady state processing overhead imposed by the fault tolerance infrastructure.

The risk categories associated with the fault tolerance infrastructure can be generally categorized as follows:

- System Performance Risk
- System Resource Usage
- System Failure Coverage

If the system is to achieve a bounded recovery time, it is necessary to employ synchronous protocols. The use of these protocols, in turn, impose strict performance requirements on such things as clock synchronization accuracy, end-to-end communications delays for critical fault tolerance messages, and event processing times.

The first priority in managing the fault tolerance infrastructure risks is to define the timing parameters and budgets required to meet the recovery time specification. Once this has been accomplished, performance modeling techniques can be used to make initial predictions and measurements of the performance of the developed code can be compared with the predictions to identify potential problem areas.

The risk management program should address such factors as the overall load imposed on the system by the fault tolerance infrastructure and the prediction and measurement of clock synchronization accuracy, end-to-end communication delays,

Although it is virtually impossible to predict the system failure coverage in advance, or verify it after-the-fact with enough accuracy to be useful, a series of risk reduction demonstrations using Government generated scenarios that attempt to "break" the fault-tolerant mechanisms has proven to be effective in exposing latent design defects in the infrastructure software. Using this approach, it is often possible that the defects can be corrected before deployment.

7.4.3.2 Application Fault Tolerance Risk Management

Monitoring the embedded fault tolerance capabilities in application software is particularly challenging because functionality, not fault tolerance, is the primary focus of the application software developers. Risk management in this area consists of:

- Establishing fault tolerance design guidelines for application developers, and
- Monitoring the compliance of the application software with the design guidelines.

The overall fault tolerance infrastructure is primarily concerned with redundancy management – that is, with monitoring the "health" of hardware and software modules and performing whatever reconfigurations and switchovers are needed to mask failures of these modules. In essence, the fault tolerance infrastructure software deals with the interaction of "black boxes."

In contrast with this basic infrastructure, application fault tolerance is intimately connected with details of the functions that the application performs and with how it interfaces with other applications. Consider a possible scenario: one application module asks another to amend a flight plan, but the receiving application has no record of that flight plan. Among the possible responses, the receiving application could simply reject the amendment, it could request that the entire flight plan be resubmitted, or it could send an error message to the controller who (it assumes) submitted the request.

What should not be allowed to happen in the above scenario would be for the error condition to propagate up to the interface between the application module and the fault tolerance infrastructure. At that level, the only way to handle the problem would be to switch to a standby application module – and that module would just encounter the same problem. Simply stated, the fault tolerance infrastructure is not equipped to handle application-specific error conditions. This high-level capability should only handle catastrophic software failures such as a module crash or hang.

The first step in effective risk management for the development of fault-tolerant application software is to establish definitive fault tolerance programming standards for the application software developers. These standards should specify different classes of faults and the manner in which they should be handled. Programmers should be required to handle errors at the lowest possible level and prohibited from simply propagating the error out of their immediate domain.

Since an application programmer's primary focus is on delivering the required functionality for their application, it will be a continuing battle to monitor their compliance with the fault tolerance programming standards. Automated tools are available that can search the source code exception handling and identify questionable exception handling practices. Failure Modes and Effects Analysis (FMEA) techniques can be used to review the error handling associated with transactions between software application modules. Traditional FMEA and Failure Mode, Effects and Criticality Analysis (FMECA) techniques such as those described in MIL-STD-1629A or System Safety practices defined in MIL-STD-

882D are oriented toward military weapons systems and are focused toward failures that directly cause injury or loss of life.

What is needed for application fault tolerance is a systematic approach to identify potential erroneous responses in the communications between software applications and verification that appropriate responses to the error conditions are incorporated into the software.

The important point to recognize is that the fault tolerance infrastructure alone cannot ensure a successful fault-tolerant system. Without "grassroots" fault tolerance embedded throughout the application software, the redundancy management fault tolerance infrastructure will be ineffective in ensuring a high reliability system.

Fault tolerance must be embedded in the applications from the ground up, as the software is developed. It can be extremely difficult to attempt to incorporate it after the fact.

The job of the application fault tolerance risk management activity is to ensure that the programmers have fault tolerance programming standards at the start of software development and to continuously track their adherence to the standards throughout the implementation phase.

7.4.3.3 Performance Monitoring Risk Management

As noted in 7.2.1.1, system performance and response times are closely coupled to reliability issues. The requirement to have rapid, consistent automatic fault detection and recovery times imposes rigid and inflexible response time requirements on the internal messages used to monitor the system's health and initiate automatic recovery actions. If the allocated response times are exceeded, false alarms may be generated and inconsistent and incomplete recovery actions will result.

Although it is the contractor's responsibility to allocate recovery time requirements to lower level system design parameters, attempting to design to unrealistic parameters can significantly increase program risk. Ultimately, it is likely that the recovery time requirement will need to be reduced to an achievable value. It is preferable, however, to avoid the unnecessary cost and schedule expenses that result from attempting to meet an unrealistic requirement. The Government should attempt to write realistic requirements. It is also necessary to watch the development closely through a contractor-developed, but Government-monitored, risk management effort. Establishing performance parameters tailored to the performance dependent RMA characteristics and formally monitoring those parameters through periodic risk management activities is an effective means of mitigating the associated risks.

7.5 Design Validation and Acceptance Testing

As discussed previously, it is not possible to verify compliance with stringent reliability requirements within practical cost and schedule constraints. There is, however, much that can be done to build confidence in the design and operation of the fault tolerance mechanisms and in the overall stability of the system and its readiness for deployment.

7.5.1 Fault Tolerance Diagnostic Testing

Despite an aggressive risk management program, many performance and stability problems do not materialize until large scale testing begins. The SAR and the DR&A capabilities provide an opportunity to leverage the data recorded during system testing to observe the operation of the fault tolerance protocols and diagnose problems and abnormalities experienced during their operation.

For system testing to be effective, the SAR and DR&A capabilities should be available when testing begins. Without these capabilities it is difficult to diagnose and correct internal software problems.

7.5.2 Functional Testing

Much of the test time at the FAATC is devoted to verifying compliance with each of the functional requirements. This testing should also include verification of compliance with the functional requirements for the systems operations functions including:

- Monitor and Control (M&C)
- System Analysis and Recording (SAR)
- Data Reduction and Analysis (DR&A)

7.5.3 Reliability Growth Testing

As discussed in Section 5.2.2.2, a formal reliability demonstration test in which the system is either accepted or rejected based on the test results is not feasible. The test time required to obtain a statistically valid sample is prohibitive, and the large number of software failures encountered in any major software development program would virtually ensure failure to demonstrate compliance with the requirements. Establishing “pass-fail” criteria for a major system acquisition is not a viable alternative.

Reliability growth testing is an on-going process of testing, and correcting failures. Reliability growth was initially developed to discover and correct hardware design defects. Statistical methods were developed to predict the system MTBF at any point in time and to estimate the additional test time required to achieve a given MTBF goal.

Reliability growth testing applied to automation systems is a process of exposing and correcting latent software defects. The hundreds of software defects exposed during system testing, coupled with the stringent reliability requirements for these systems, preclude the use of statistical methods to accurately predict the test time to reach a given MTBF prior to system deployment. There is no statistically valid way to verify compliance with reliability requirements at the FAATC prior to field deployment. There is a simple reason for this: it is not possible to obtain enough operating hours at the FAATC to reduce the number of latent defects to the level needed to meet the reliability requirements.

The inescapable conclusion is that it will be necessary to field systems that fall short of meeting the reliability requirements. The large number of additional operating hours accumulated by multiple system installations will increase the rate that software errors are found and corrected and the growth of the system MTBF.

To be successful, the reliability growth program must address two issues. First, the contractor must be aggressive at promptly correcting software defects. The contractor must be given a powerful incentive to keep the best people on the job through its completion, instead of moving them to work on new opportunities. This can be accomplished by a process called “expunging.” The system MTBF was computed by dividing the operating hours by the number of failures. However if the contractor could demonstrate that the cause of the failure had been corrected then the failure was “expunged” from the list of failures. If a failure cannot be repeated within 30 days, it is also expunged from the database.

Thus, if all Program Trouble Reports (PTRs) are fixed immediately, the computed MTBF would be infinity even if the system were failing on daily basis. This measure is statistically meaningless as a true indicator of the system MTBF. It is, however, a useful metric for assessing the responsiveness of the contractor in fixing the backlog of accumulated PTRs. Since the Government representatives decide when to expunge errors from the database, they have considerable leverage over the contractor by controlling

the value of the MTBF reported to senior program management officials. There may be other or better metrics that could be used to measure the contractor's responsiveness in fixing PTRs. The important thing is that there must be a process in place to measure the success contractor's support of reliability growth.

The second issue that must be addressed during the reliability growth program is the acceptability of the system to field personnel. In all probability, the system will be deployed to field sites before it has met the reliability requirements. Government field personnel should be involved in the reliability growth testing at the FAATC and concur in the decision concerning when the system is sufficiently stable to warrant sending it to the field.

As discussed in Section 5.2.2.2, it is not possible to verify compliance with stringent reliability requirements within practical cost and schedule constraints. There is, however, much that can be done to build confidence in the design and operation of the fault tolerance mechanisms and in the overall stability of the system and its readiness for deployment. The way to accomplish this is to provide the test tools, personnel, and test time to pursue an aggressive reliability growth program.

8 NAS-SR-1000 MAINTENANCE

Clearly, if theNAS-SR-1000 is to be effective in guiding the evolution of the NAS Architecture, it will have to be a living document. The RMA requirements have been designed so that, with the exception of the Service Threads, they should be largely independent of changes in the NAS Architecture or theNAS-SR-1000 functional requirements. The basic concepts of criticalities associated with functions and the RMA requirements associated with those criticalities should remain relatively constant.

8.1 Revising Service Thread Requirements

One of the advantages of the Service Thread based approach is that the Service Threads can remain relatively constant as the NAS Architecture evolves. Many, if not most, of the changes to the NAS Architecture involve replacement of a facility representing a block in the reliability block diagram for the thread. Thus, the basic thread does not need to change, only the name of a block in the thread. As the NAS evolves, the Service Thread Diagrams should evolve with it.

The NAS Infrastructure Diagram uses a color-coded legend to distinguish between operational systems, systems being installed, and prototyping systems. A similar methodology would be useful for the blocks in the Service Thread reliability block diagrams. Because the Service Threads need to address requirements and specification issues long before prototyping and deployment, an additional category for systems approved for acquisition may be needed.

8.2 Adding a New Service Thread

While the addition of a new Service Thread to the NAS is a relatively rare occurrence, and has not happened within the past five years, Service Threads may need to be added in the future to accommodate new NAS capabilities. Provisions should be made so that it is not overly difficult to make these additions. Maintaining a flexible approach to Service Thread mapping will facilitate the accommodation of new threads when they are needed.

With the structure provided by the RMA Requirements Document, new Service Threads will not necessarily have to rely on the NAS-Level RMA criticality definitions. Rather, it will be possible to move straight to defining the SLTSC for the new Service Thread and its place in the NAS. Then, all RMA-related requirements should follow more easily.

There are two instances in which new Service Threads will need to be added to the existing set:

- To accommodate services such as navigation that are provided to pilots but are not identified in the current FAA order 6040.15D.
- To add new Service Threads corresponding to new NAS capabilities that are not reflected in the current set of 6040.15D services.

A major objective of this effort was to couple the RMA requirements to real-world NAS services. The FAA Order 6040.15 services serve as the foundation for this effort. Since there will be a need to create services in addition to those defined in FAA Order 6040.15D, it will be important to continue to distinguish between the official operational FAA services and services that have been created to support requirements development and acquisition planning.

Many of the Service Threads that will need to be added will be represented by a single facility, as defined in FAA Order 6040.15D. In these cases, the Service Threads will be assigned the appropriate 6040.15D facility identification, and color-coding will distinguish them from the basic set of Service Threads.

In cases where Service Threads have to be invented during system engineering or acquisition planning, the new Threads will have a unique identifier to distinguish them from the official services defined in FAA Order 6040.15D.

Ideally, all deviations from the set of Service Threads set forth for the approved services in FAA Order 6040.15D should be coordinated with ATO Technical Operations. With this coordination, as new services are deployed, there can be an orderly transition between the hypothetical Service Threads and the FAA Order 6040.15.

TABLE 6-2 defines the mapping between the approved set of NAPRS services defined in FAA Order 6040.15D and the set of NAS-SR-1000 Service Threads. The complete set of Service Threads consists of most, but not all, of the NAPRS services, NAPRS facilities that have been converted to Service Threads, and newly created Service Threads that are not defined in FAA Order 6040.15D.

9 RMA REQUIREMENTS ASSESSMENT

The NAS-SR-1000 RMA requirements have been rewritten to allocate to Service Threads that are based on the National Airspace Performance Reporting System (NAPRS) services defined in FAA Order 6040.15. The Service Thread approach applies the NAS-Level requirements to real-world services and facilities that are precisely defined and well-understood in the engineering as well as operational communities in the FAA.

Several benefits accrue from using this approach, including the ability to close the loop between the measured RMA characteristics of operational services and systems and the NAS-Level requirements for these systems. Previously, the only real feedback reconciling RMA requirements with the actual performance of systems has been part of the WJHTC testing of newly developed systems. At this time, the Technical Center staff routinely attempt to verify compliance with the system-level specifications, as illustrated in FIGURE 9-1. With this feedback loop, however, it often proves too costly or time consuming to verify compliance of high availability systems with RMA requirements to any level of statistical significance. About the best that can be done is to demonstrate that the system has achieved suitable stability for field deployment and continue to collect reliability information in the field. With many systems in the field, the rate of exposing (and correcting) latent software defects increases and the software reliability growth rate increases.

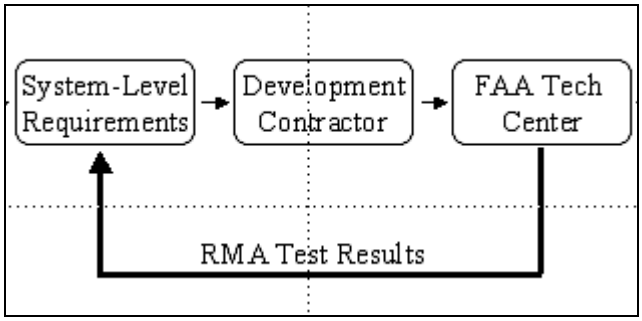

FIGURE 9-1: RMA Feedback Path

Once systems have been deployed and accepted for operational use, data on their RMA characteristics is collected through the National Airspace Reporting System (NAPRS), the official service established to provide insight into the performance of fielded RMA systems.

To sum up, before the introduction of Service Threads, there has been no satisfactory way to relate the NAS-Level requirements to the performance of existing systems. The use of Service Threads based on the NAPRS services defined in FAA Order 6040.15D as a basis for the NAS-SR-1000 RMA requirements now allows the requirements to be compared with the performance of existing systems.

The availabilities assigned to Service Threads provide a second feedback loop from the NAPRS field performance data to the NAS-Level requirements, as shown in FIGURE 9-2. This redundancy provides a mechanism for verifying the realism and achievability of the requirements, and helps to ensure that the requirements for new systems will be at least as good as the performance of existing systems

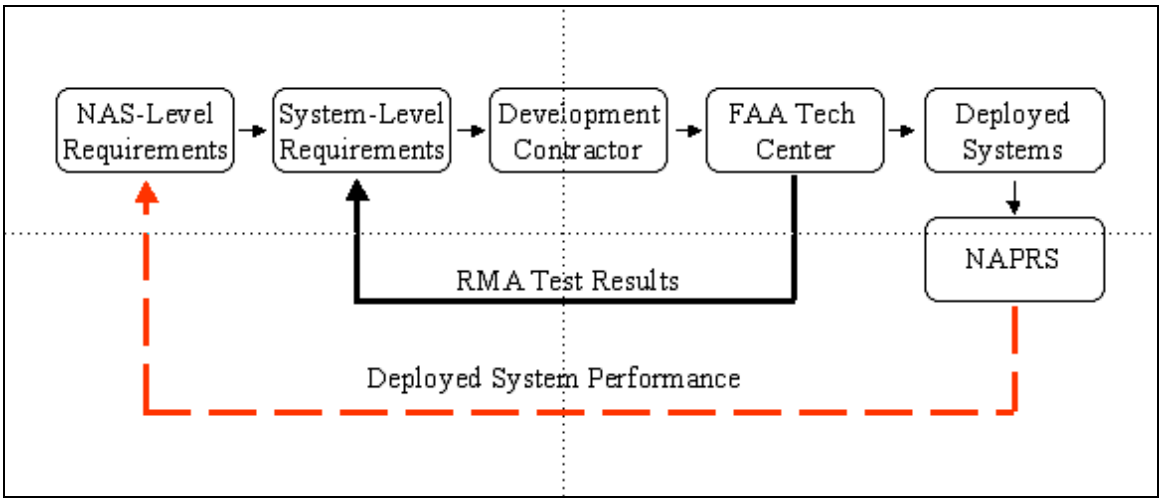

FIGURE 9-2: Deployed System Performance Feedback Path

Closing the loop provides two benefits, it allows system engineers to:

- Check realism of requirements and identify operational deficiencies.
- Look at overall characteristics of the current NAS Architecture and identify weak spots and/or areas where financial resources are not being allocated properly.

A histogram showing the distribution of the equipment and service availabilities of Service Threads for operationally deployed systems as shown in FIGURE 9-3 presents operational data for five years from FY 2000 through FY 2005. The histogram reports the number of input values that are equal to, or greater than, the bin value – but still less than the next bin value – and displays it in the Frequency column. The last value in the table reports the number of input values equal to, or greater than, the last bin value. The figure shows that most of the Service Thread availabilities are in the .999 to .9999 range.

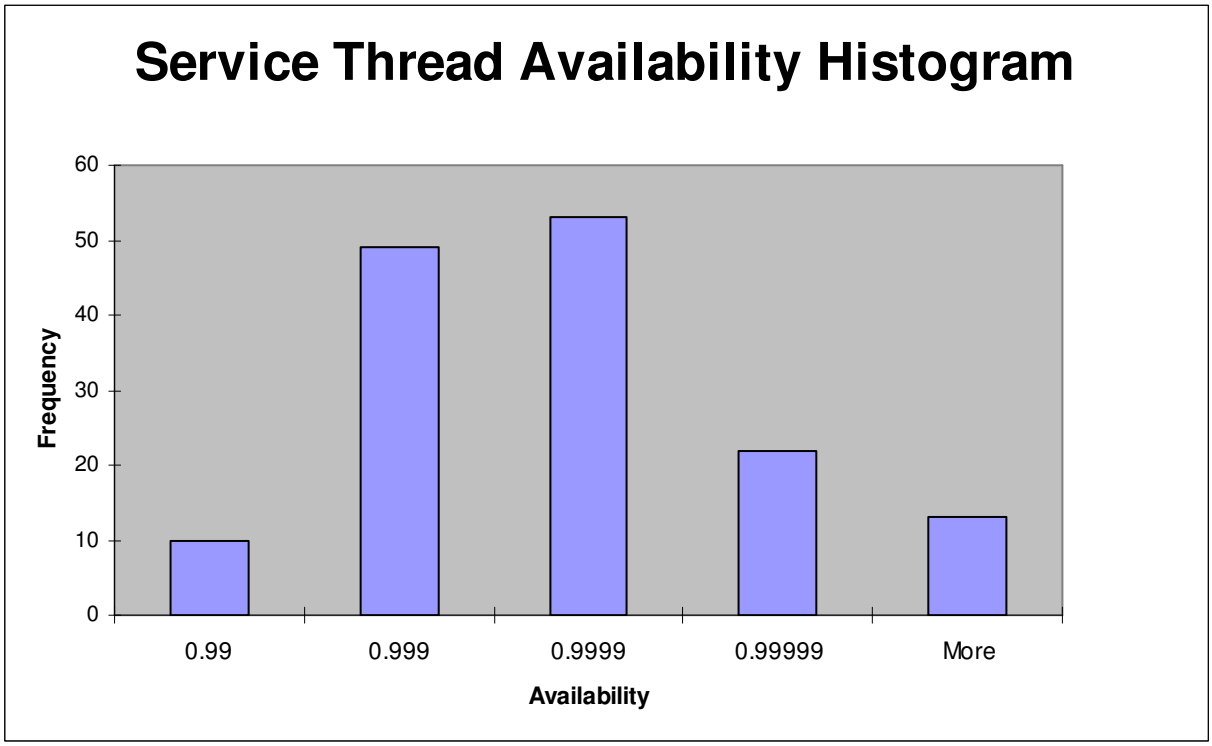

FIGURE 9-3: Service Thread Availability Histogram

FIGURE 9-4 illustrates the mean time between unscheduled interruptions for the same period. Most of the Mean Time between Outage (MTBO) values fall in the range of 1,000 to 50,000 hours, although the values range to more than 100,000 hours. A significant number of data points are greater than 100,000 hours. The average MTBO for all facilities is 52,000 hours, while the median is only 16,000 hours. A cursory examination of the raw data indicates that most of the facilities with MTBOs below 10,000 hours are older facilities, while newly acquired systems are generally above 30,000 hours. The few automation facilities with MTBOs below 10,000 hours tend to obsolete systems such as the CDC and DARC that have been or are being replaced.

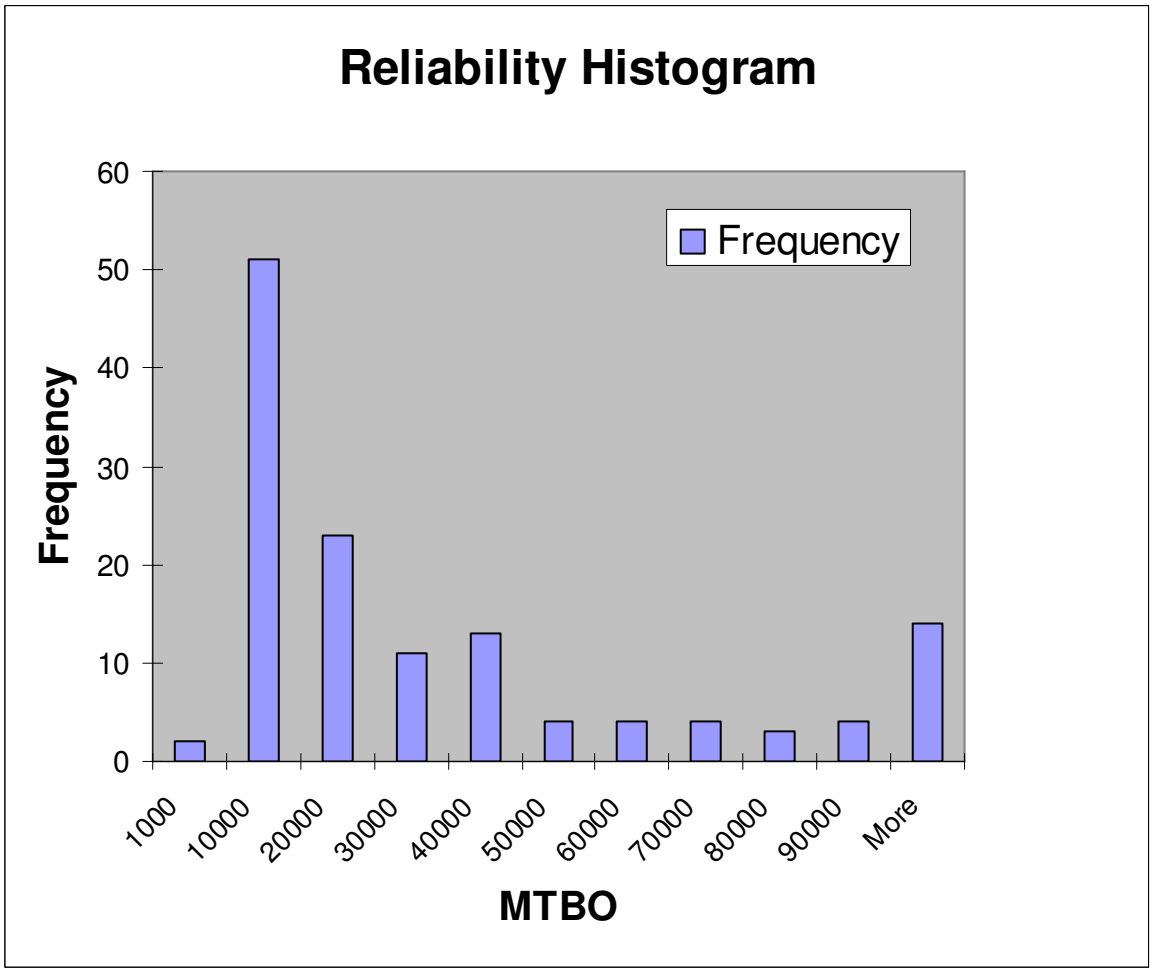

FIGURE 9-4: Reliability Histogram for Unscheduled Interruptions

9.1 Requirements Analysis

The block labeled “NAS Level Requirements” in FIGURE 9-2 has been expanded in FIGURE 9-5 to illustrate the process and considerations used to assess the reasonableness of the NAS-Level Service Thread RMA requirements.

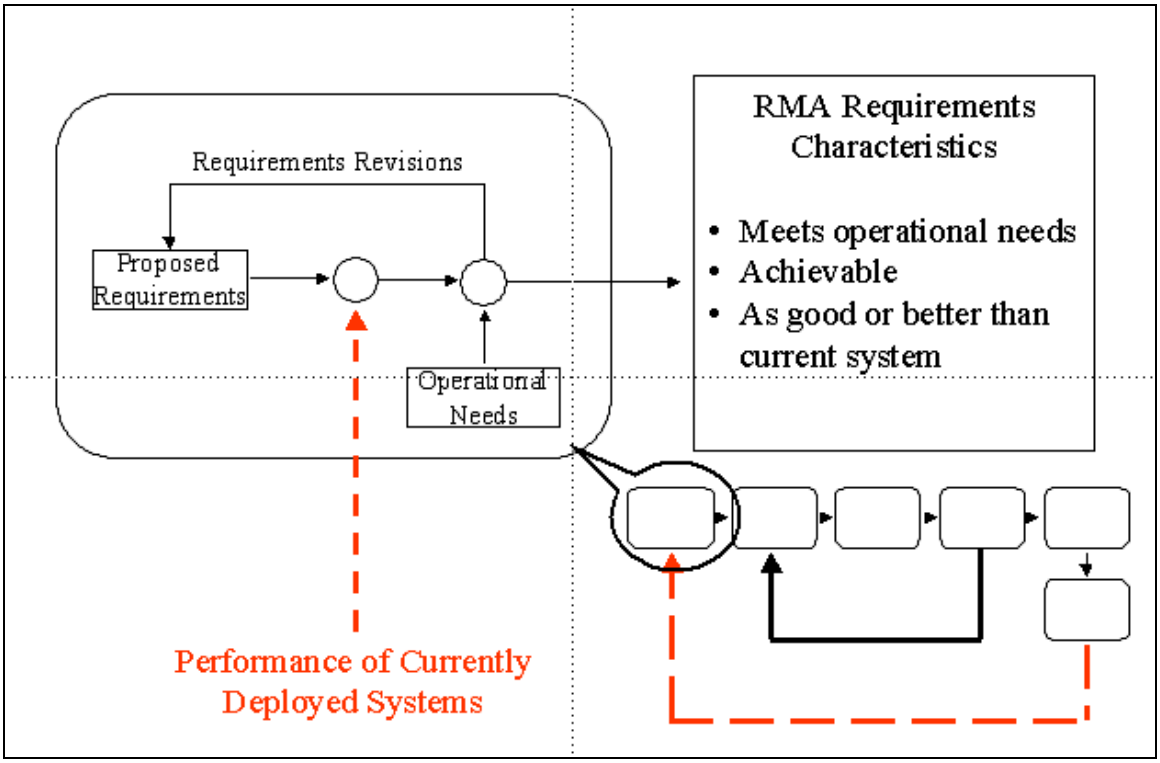

FIGURE 9-5: Requirements Analysis

Tentative RMA requirements are compared with the performance of currently fielded systems as measured by NAPRS. If the proposed requirements are consistent with the performance of currently fielded systems, then the requirements can be assumed to be realistic.

If the performance of currently fielded systems exceeds the proposed requirements, the principle that new systems being acquired must be at least as good as the systems they are replacing dictates that the RMA requirements must be made more stringent.

On the other hand, if the proposed new requirements significantly exceed the performance of existing systems, the requirements either are unrealistically stringent or the fielded systems are not performing in an operationally acceptable manner. The fact that the requirements are not consistent with the observed performance of existing systems is not, *per se,* an unacceptable situation. The motivation for replacing existing systems is often that the reliability of these systems has deteriorated to the point where their operational suitability is questionable, or the cost to maintain them has become excessive.

The operational suitability of the existing systems must be considered when proposed requirements are being evaluated.

9.2 Architecture Assessment

Another benefit of using the Service Thread approach is that, through use of the NAPRS system, it readily supports closed loop corrective action systems, such as Failure Reporting, Analysis, and Corrective Action System (FRACAS) or Data Reporting, Analysis, and Corrective Action System (DRACAS), that can be used to assess the NAS Architecture. The additional feedback path is illustrated in FIGURE 9-6. This data can support the analysis of the contributions of the components of a Service Thread to the overall reliability of the service. The objective of this analysis process is to work toward improving the

overall reliability of the NAS Architecture by identifying the weak links and applying resources to those areas that will have the greatest potential for improving the overall NAS reliability. For example, if analysis shows that the predominate cause interruptions of surveillance services is the failure of communications links or power interruptions, then attempting to acquire highly reliable radar or surveillance processing systems alone will not improve the overall reliability of surveillance services. The analysis of field data can assist system engineers in focusing on those areas of the NAS Architecture that offer the greatest opportunity for improving the reliability of NAS services.

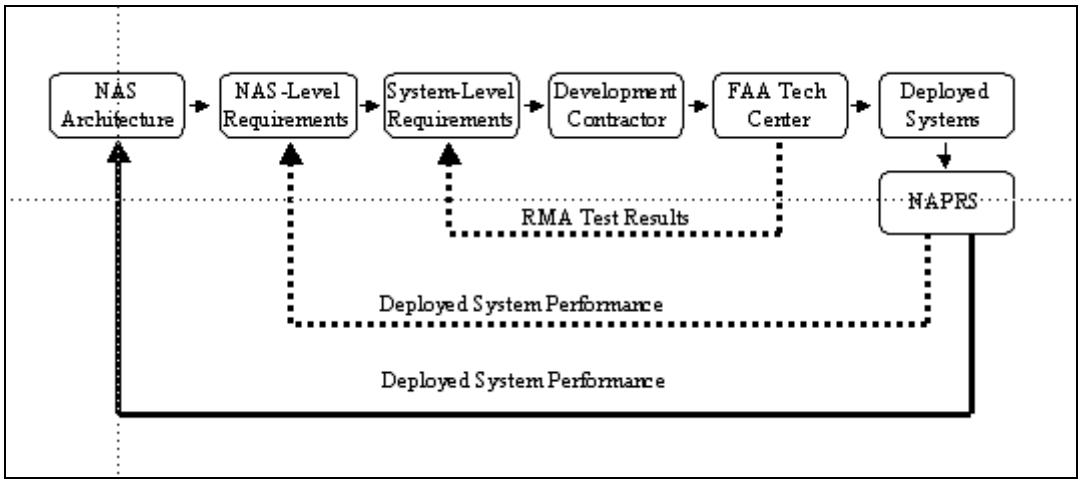

FIGURE 9-6: Architecture Assessment

10 NOTES

10.1 Updating this Handbook

This handbook is designed to be a living document that will develop and be refined over time, both through changes in the NAS and theNAS-SR-1000, and through its use to assist in the preparation of RMA packages for system acquisitions. While the first process will be driven by FAA Systems Engineering, the second process will only be possible if the users of the Handbook comment on its use.

While the handbook is being used for its intended purpose, the acquisition manager or Business Unit personnel should keep notes regarding the areas where the Handbook was either helpful or where it was lacking. These notes and comments about the Handbook should then be provided to the FAA Systems Engineering Office, NAS Requirements and Interface Management Division, the FAA Systems Engineering, so that they can be incorporated into future revisions of the Handbook.

10.2 Bibliography

The following sources contain information that is pertinent to the understanding of the RMA related issues discussed in this handbook. Reviewing these resources will equip the user of this handbook with the necessary background to develop, interpret and monitor the fulfillment of RMA requirements.

Abouelnaga, Ball, Dehn, Hecht, and Sievers. Specifying Dependability for Large Real-Time Systems. *1995 Pacific Rim International Symposium on Fault-Tolerant Systems (PRFTS)*, December 1995.

Avizienis, Algirdas and Ball, Danforth. On the Achievement of a Highly Dependable and Fault-Tolerant Air Traffic Control System. *IEEE Computer*, February 1987.

Ball, Danforth. User Benefit Infrastructure: An Integrated View. *MITRE Technical Report MTR 95W0000115*, September 1996.

Ball, Danforth. COTS/NDI Fault Tolerance Options for the User Benefit Infrastructure. *MITRE Working Note WN 96W0000124*, September 1996.

Brooks, Frederick. The Mythical Man-Month: *Essays on Software Engineering*. Wesley Publishing, 1975.

DeCara, Phil. En Route Domain Decision Memorandum.

En Route Automation Systems Supportability Review, Volume I: Hardware, October 18, 1996.

En Route Automation Systems Supportability Review, Volume II: Software & Capacity, February 12, 1997.

Hierro, Max del. Addressing the High Availability Computing Needs of Demanding Telecom Applications. VMEbus Systems, October 1998.

IEEE Std 493-1997, Recommended Practice for the Design of Reliable Industrial and Commercial Power Systems, (Gold Book).

Mills, Dick. Dancing on the Rim of the Canyon, August 21, 1998

Resnick, Ron I. A Modern Taxonomy of High Availability. http://www.interlog.com/~resnick/ron.html, December 1998

Talotta, Michael. Software Diversity Study for EDARC Replacement, May 1997.

Voas, Ghosh, Charron, and Kassab. Reducing Uncertainty about Common Mode Failures. *Pacific Northwest Software Quality Conference*, October 1996.

Voas, Jeffery and Kassab, Lora. Simulating Specification Errors and Ambiguities in Systems Employing Design Diversity. http://www.itd.nrl.navy.mil/ITD/5540/publications/CHACS/1997/1997kassab-PNSQ97.pdf, December 1998.

Wellman, Frank. Software Costing. Prentice Hall, 1992.

10.3 Other Notes

Appendix A SAMPLE REQUIREMENTS

This appendix presents sample requirements that the reader may find useful in developing System Level Specifications and other procurement documents. The reader is cautioned that these checklists contain requirements that may not be applicable to every system. Some of the requirements may need to be tailored to a specific system. The requirements are organized around the documents and paragraphs of those documents where they are most applicable. The numbers in parentheses, e.g., (3.7.1.A) (~18190-18220), following the sample requirements provide cross references to the SR-1000 requirements from which they were derived. Numbers not preceded by a tilde "~" refer to the March 1995 version of SR-1000; numbers preceded by a tilde refer to the SR-1000A version that is based on the NAS Architecture.

The standard outline for System Level Specifications has three separate paragraphs for requirements related to RMA: System Quality Factors, System Design Characteristics and System Operations. The paragraphs below present sample requirements for each of the three sections.

A.1 System Quality Factors

System Quality Factors include those requirements associated with attributes that apply to the overall system. They typically include requirements for Availability, Reliability and Maintainability.

Availability Requirements – The following table presents potential availability requirements.

Potential Availability Quality Factor Requirements
The system shall have a minimum inherent availability of (*). (3.8.1.B)[14] * This value is determined by referencing TABLE 6-5: Service Thread Reliability, Maintainability, and Recovery Times.

Reliability Requirements – The following table presents potential availability requirements.

Potential Reliability Quality Factor Requirements
The predicted Mean Time Between Failures (MTBF) for the system shall be not less than (*) hours. * This value is determined by referencing TABLE 6-5: Service Thread Reliability, Maintainability, and Recovery Times. The reliability of the system shall conform to Table X (Reliability Growth Table).

Maintainability Requirements – The following table presents potential maintainability requirements.

Potential Maintainability Quality Factor Requirements
The mean time to repair (MTTR) for all equipment shall be 30 minutes or less.

[14] Parenthetical references are to the NAS-SR-1000a.

The mean time to restore service (MTTRS) shall be 30 minutes or less.

The maximum time to restore service shall be 120 minutes or less for failed Floor Replaceable Units (FRUs) and Lowest Replaceable Units (LRUs).

The maximum time to restore service shall be 8 hours or less for Maintenance Significant Items (MSIs).

Restoral times service shall include diagnostic time (fault isolation), removal of the failed Lowest Replaceable Units (LRU), Floor Replaceable Units (FRU), or Maintenance Significant Items (MSI) replacement and installation of the new LRU, FRU, or MSI including any adjustments or data loading necessary to initialize the LRU, FRU, or MSI (including any operating system and/or application software), all hardware adjustments, verifications, and certifications required to return the subsystem to normal operation, and repair verification assuming qualified repair personnel are available and on-site when needed.

Subsystem preventive maintenance shall not be required more often than once every three months. (3.7.1.)

Preventive maintenance on any subsystem shall not require more than 2 staff hours of continuous effort by one individual

A.2 System Design Characteristics

System Design Characteristics related to RMA – The following table presents potential system availability related design characteristics.

Potential Availability System Design Characteristics

The system shall have no single point of failure. (3.8.1.C)

Reliability Design Characteristics – The following table presents potential system reliability related design characteristics.

Potential Reliability Design Characteristics

The system shall restart without requiring manual reentry of data.

Where redundant hardware or software is used to satisfy reliability requirements, the system shall automatically switchover from a failed element to the redundant element.

Where redundant hardware or software is used to satisfy reliability requirements, the system shall monitor the health of all redundant elements.

Maintainability Design Characteristics – The following table presents potential maintainability design characteristics.

Potential Maintainability Design Characteristics
The system shall support scheduled hardware maintenance operations without increasing specialist workload.
The system shall support scheduled software maintenance operations without increasing specialist workload.
The system shall enable field level technical personnel to correct equipment failures by replacing faulty Lowest Replaceable Unit (LRUs) and Floor Replaceable Unit (FRUs).
The system shall permit the technician to physically remove and replace a Floor Replaceable Unit (FRU) diagnosed within (TBD) minutes.
The system shall permit replacement of any Lowest Replaceable Unit (LRU) while all functional operations continue uninterrupted on redundant equipment.
The system shall permit replacement of any Floor Replaceable Unit (FRU) while all functional operations continue uninterrupted on redundant equipment.
The system shall permit replacement of any Maintenance Significant Item (MSI) while all functional operations continue uninterrupted on redundant equipment.
[Optional, for systems employing multiple, independent data paths] Maintenance operations performed on a single data path shall not impact operations on the alternate data path.

A.3 System Operations

Maintainability Functional Requirements – The following table presents potential maintainability functional requirements.

Potential Maintainability Functional Requirements
Failed resources shall be isolatable from the system for performance of maintenance operations.
System elements shall require no more than one hour of Periodic Maintenance (PM) to less than one hour per year for each element, subsystem and their respective Lowest Replaceable Unit (LRUs) and Floor Replaceable Unit (FRUs) excluding any mechanical devices (such as printers).
All Lowest Replaceable Unit (LRUs) shall be accessible and removable at the equipment's operational location.
All Floor Replaceable Unit (FRUs) shall be accessible and removable at the equipment's

operational location.

All Maintenance Significant Item (MSIs) shall be accessible and removable at the equipment's operational location.

The system shall be available for operational use during routine tasks:

- *Maintenance*
- *Hardware diagnostics*
- *Software diagnostics*
- *Verification testing*
- *Certification testing*
- *Training*

The system shall provide for the building and implementing of specific databases.

The system shall provide for identifying software problems. (3.7.1.A) (~18190-18220)

The system shall provide for identifying hardware problems. (3.7.1.A) (~18190-18220)

The system shall provide for collecting support data. (3.7.1D.2) (~18900-18910)

The system shall provide for displaying problem description data. (3.7.1.D.2.c) (~18870)

The system shall receive software versions from selected software support sites.

The system shall reload a selected software version from a storage device.

The system shall test that the version or modification to existing software meets requirements for operational use. (3.7.1.C.1.c) (~18450)

The system shall verify that the version or modification to existing software meets requirements for operational use. (3.7.1.B) (~18320)

The system shall validate that the version or modification to existing software meets requirements for operational use. (3.7.1.B) (~18320)

The system shall accept new operational software.

The system shall accept new maintenance software.

The system shall accept new test software.

The system shall accept new training software.

Monitor and Control – The following table presents potential Monitor and Control (M&C) General functional requirements.

Potential M&C General Functional Requirements

The system shall have a Monitor and Control (M&C) function. (3.7.1.A) (~17880-18000)

Specialists shall be provided with a means to interact with the M&C function via M&C commands. (3.7.1.C.3.a)

The Monitor and Control (M&C) function shall monitor system health. (3.7.1.A) (~17890, 17970-17980)

The Monitor and Control (M&C) function shall monitor system performance. (3.7.1.A) (~17900)

The Monitor and Control (M&C) function shall control system configuration. (3.7.1.A) (~17990-18000)

[Optional, for systems with multiple data paths.] The Monitor and Control (M&C) function shall support verification and certification of one data path while the other data path supports normal operation. (3.7.1.B) (~18320) (3.7.1.C.1.c) (~18450)

Upon Monitor and Control (M&C) command, The M&C function shall, create a hard copy printout of specialist-selected textual output, including displayed status and error messages.

The system shall continue operations without interruption whenever one or more M&C Positions fail.

The system shall perform automatic recovery actions in response to the failure of any hardware or software component without reliance on the Monitor and Control (M&C) function.

Upon Monitor and Control (M&C) command, the M&C function shall restore applications databases after restart from internal recovery.

Upon Monitor and Control (M&C) command, the M&C function shall restore applications databases after restart by reconstitution from external sources.

Upon Monitor and Control (M&C) command, the M&C function shall test non-operational assemblies and identify failed assemblies to the LRU and FRU level without any degradation to normal operations. (3.7.1.A.1.b) (~18060)

Upon Monitor and Control (M&C) command, the M&C function shall initiate off-line diagnostics to test and isolate an indicated fault in an LRU without the use of operational equipment. (3.7.1.A.1.b) (~18060)

Upon Monitor and Control (M&C) command, the M&C function shall initiate off-line diagnostics to test and isolate an indicated fault in an FRU without the use of operational equipment. (3.7.1.A.1.b) (~18060)

Upon Monitor and Control (M&C) command, the M&C function shall initiate off-line diagnostics to test and isolate an indicated fault in an MSI without the use of operational

equipment. (3.7.1.A.1.b) (~18060)

The system shall automatically recover from a power outage.

The system shall automatically recover from a software fault.

System Monitoring Functional Requirements – The following table presents potential M&C System Monitoring functional requirements.

Potential M&C System Monitoring Functional Requirements

The Monitor and Control (M&C) function shall monitor all critical parameters required to determine the operational status of each software component of the system. (3.7.1.A) (~17880-17890)

The Monitor and Control (M&C) function shall collect equipment status data. (3.7.1.A) (~17890)

The Monitor and Control (M&C) function shall collect equipment performance data. (3.7.1.A) (~17890)

The Monitor and Control (M&C) function shall display equipment status data. (3.7.1.A) (~17890)

The Monitor and Control (M&C) function shall display equipment performance data. (3.7.1.A) (~17890)

The Monitor and Control (M&C) function shall monitor parameters required to determine the operational status of each hardware component of the system, at a minimum to the Lowest Replaceable Unit (LRU) level. (3.7.1.A) (~17880-17890)

The Monitor and Control (M&C) function shall monitor parameters required to determine the operational status of each external system interface. (3.7.1.A) (~17880-17890)

The Monitor and Control (M&C) function shall monitor parameters required to determine the current system configuration. (3.7.1.A) (~17880-17890)

The Monitor and Control (M&C) function shall monitor parameters required to determine the current hardware identification configuration. (3.7.1.A) (~17880-17890)

The Monitor and Control (M&C) function shall monitor parameters required to determine the current software identification configuration. (3.7.1.A) (~17880-17890)

The Monitor and Control (M&C) function shall monitor parameters required to determine the configuration of all reconfigurable resources. (3.7.1.A) (~17880-17890)

[Optional for systems with multiple data paths] The M&C function shall monitor parameters required to determine which data path has been selected by each operational position. (3.7.1.A) (~17880-17890)

The Monitor and Control (M&C) function shall monitor parameters required to derive the status of the M&C position. (3.7.1.A) (~17880-17890)

The Monitor and Control (M&C) function shall monitor parameters required to derive the availability status of each operational function of the system. (3.7.1.A) (~17880-17890)

The Monitor and Control (M&C) function shall monitor parameters required to derive the availability status of each operational support function of the system. (3.7.1.A) (~17880-

17890)

The Monitor and Control (M&C) function shall monitor parameters required to derive the system-level status of the system. (3.7.1.A) (~17880-17890)

The Monitor and Control (M&C) function shall monitor parameters required to certify the system. (3.7.1.A) (~17880-17890)

The Monitor and Control (M&C) function shall record system performance parameters at every (TBD seconds). (3.7.1.D) (~18820)

The Monitor and Control (M&C) function shall perform system performance data collection while meeting other performance requirements. (3.7.1.A) (~17900)

The Monitor and Control (M&C) function shall perform system performance data collection without the need for specialist intervention. (3.7.1.A) (~17900)

The Monitor and Control (M&C) function shall determine the alarm/normal condition and state change events of all sensor parameters, derived parameters, ATC specialist positions, M&C positions, system functions and subsystem operations.

The Monitor and Control (M&C) function shall determine the alarm/normal condition and state change events of all sensor parameters. (3.7.1.A.3.a) (~17960)

The Monitor and Control (M&C) function shall determine the alarm/normal condition and state change events of all derived parameters. (3.7.1.A.3.a) (~17960)

The Monitor and Control (M&C) function shall determine the alarm/normal condition and state change events of all specialist positions. (3.7.1.A.3.a) (~17960)

The Monitor and Control (M&C) function shall determine the alarm/normal condition and state change events of all M&C positions. (3.7.1.A.3.a) (~17960)

The Monitor and Control (M&C) function shall determine the alarm/normal condition and state change events of all system functions. (3.7.1.A.3.a) (~17960)

The Monitor and Control (M&C) function shall determine the alarm/normal condition and state change events of all subsystem operations. (3.7.1.A.3.a) (~17960)

The Monitor and Control (M&C) function shall provide state change comparisons as part of status determination. (3.7.1.A.3.a) (~17960)

The M&C function shall report status notifications to the M&C position without specialist intervention. (3.7.1.A.3.a) (~17960)

The Monitor and Control (M&C) function shall display alarm notifications to the M&C position within (TBD seconds of their occurrence. (3.7.1.A.3.a) (~17960)

The Monitor and Control (M&C) function shall include the monitored parameter associated with an alarm notification.

The Monitor and Control (M&C) function shall include the date and time that a condition was

declared with reporting/displaying an alarm condition.

The Monitor and Control (M&C) function shall display system state changes to the M&C position within (TBD seconds). (3.7.1.A.3.a) (~17960)

The Monitor and Control (M&C) function shall include the monitored parameter associated with a state change occurrence in a state change notification.

The Monitor and Control (M&C) function shall include the date and time that a condition was declared with a state change.

The Monitor and Control (M&C) function shall display return-to-normal notifications to the M&C position within (*) seconds. (3.7.1.A.3.a) (~17960) *Value to be supplied by the Business Unit.

The Monitor and Control (M&C) function shall include the monitored parameter associated with a return-to-normal condition in a return-to-normal notification.

The Monitor and Control (M&C) function shall include the date and time that a condition was declared with return-to-normal condition.

All generated alarm/return-to-normal/state change notifications shall be retained in a form which allows on-line specialist-selectable retrieval for a period of at least (*) hours. (3.7.1.A.3.c) (~18310) *Value to be supplied by the Business Unit.

The Monitor and Control (M&C) function shall display specific monitored parameters when requested by the M&C position. (3.7.1.A) (~17900)

The Monitor and Control (M&C) function shall continually monitor: [Include a specific requirement for each that applies.] (3.7.1.A) (~17900)

- *network and network component utilization*
- *processor utilization*
- *input/output peripheral attachment path utilization*
- *peripheral device utilization*
- *memory page fault rates*
- *memory utilization*
- *software utilization*
- *operating system parameters*

The Monitor and Control (M&C) function shall display a selected set of monitored parameters when requested by the M&C position. (3.7.1.A) (~17900)

The Monitor and Control (M&C) function shall display the most recently acquired monitor parameters in performance data reports. (3.7.1.A) (~17900)

The Monitor and Control (M&C) function shall display the most recently determined

alarm/normal conditions in performance data reports. (3.7.1.A) (~17900)

The Monitor and Control (M&C) function shall display subsystem status when requested by the M&C position. (3.7.1.A) (~17900)

The Monitor and Control (M&C) function shall display control parameters when requested by the M&C position. (3.7.1.A) (~17900)

All reported parameters shall be logically grouped according to subsystem structure.

Each reported parameter logical grouping shall be uniquely identifiable.

Each reported parameter within a logical grouping shall be uniquely identifiable.

System Control – The following table presents potential M&C Control Functional requirements.

Potential M&C System Control Functional Requirements

The M&C function shall support initializing the system. (3.7.1.A.1) (~17990-1800)

The M&C function shall support startup of the system. (3.7.1.A.1) (~17990-1800)

The M&C function shall support restarting the system with recovery data. (3.7.1.A.1) (~17990-1800)

The M&C function shall support restarting the system without recovery data. (3.7.1.A.1) (~17990-1800)

The M&C function shall support the option of restarting individual processors with recovery data. (3.7.1.A.1) (~17990-1800)

The M&C function shall support the option of restarting individual processors without recovery data. (3.7.1.A.1) (~17990-1800)

The M&C function shall support the option of restarting individual consoles with recovery data. (3.7.1.A.1) (~17990-1800)

The M&C function shall support the option of restarting individual consoles recovery data. (3.7.1.A.1) (~17990-1800)

The M&C function shall control the shutdown of the system. (3.7.1.A.1) (~17990-1800)

The M&C function shall control the shutdown of individual processors. (3.7.1.A.1) (~17990-1800)

The M&C function shall control the shutdown of individual consoles. (3.7.1.A.1) (~17990-1800)

The M&C function shall control the loading of new software releases into system processors. (3.7.1.A.1) (~17990-1800)

The M&C function shall have the capability to control the cutover of new software releases in system processors. (3.7.1.A.1) (~17990-1800)

The M&C function shall have the capability to control the cutover of prior releases in system processors. (3.7.1.A.1) (~17990-1800)

The M&C function shall control the initiating of the System Analysis Recording (SAR) function. (3.7.1.A.1) (~17990-1800)

The M&C function shall control the stopping of the System Analysis Recording (SAR) function. (3.7.1.A.1) (~17990-1800)

The M&C function shall control what data is recorded by the System Analysis Recording (SAR) function. (3.7.1.A.1) (~17990-1800)

The M&C function shall enable/disable the alarm/normal detection of monitored parameters. (3.7.1.A.1) (~17990-1800)

M&C Computer/Human Interface (CHI) Requirements – The following table presents potential M&C CHI requirements.

Potential M&C CHI Requirements

[If applicable] All M&C position displays shall be presented to the specialist in the form of movable, resizable windows.

The M&C function shall provide a set of views that allow the specialist to "drill down" to obtain increasingly detailed performance and resource status.

The M&C function shall simultaneously display a minimum of (TBD) displays on the same workstation, with no restrictions as to display content.

The M&C function shall display an applicable error message if an invalid request or command is entered.

The M&C function shall display graphical information using redundant information coding [e.g. color, shapes, auditory coding] to highlight resource status.

The M&C function shall display list information using redundant information coding (e.g. color, shapes, auditory coding) to highlight resource status.

The M&C function shall support command composition using a combination of keyboard entries and pointer device selections.

The M&C function shall support command initiation using a combination of keyboard entries and pointer device selections.

The M&C function shall display commands under development for confirmation prior to execution.

The M&C function shall initialize all specialist-modifiable system parameters to default values.

The M&C function shall provide consistent and standardized command entry such that similar actions are commanded in similar ways.

The M&C function shall prevent inadvertent or erroneous actions that can degrade operational capability.

M&C function generated messages shall be presented in concise, meaningful text, such that the translation of error, function, or status codes is not required of the specialist in order to understand the information.

M&C function generated alerts shall be presented in concise, meaningful text, such that the translation of error, function, or status codes is not required of the specialist in order to

understand the information.

M&C function generated warnings shall be presented in concise, meaningful text, such that the translation of error, function, or status codes is not required of the specialist in order to understand the information.

M&C function generated visual alarms shall warn of errors, out of tolerance conditions, recovery actions, overloads, or other conditions that may affect system operation or configuration. [Include individual requirements for each that applies.] (3.7.1.A.3) (~17960 and 18450)

- *Errors*
- *Out of tolerance conditions*
- *Recovery action*
- *Overloads*

M&C function generated aural alarms shall warn of conditions that may affect system operation or configuration. [Include individual requirements for each that applies.] (3.7.1.A.3) (~17960 and 18450)

- *Errors*
- *Out of tolerance conditions*
- *Recovery action*
- *Overloads*

M&C function generated visual alarms shall be designed to incorporate clearly discriminative features which distinguish the warning (e.g., color, blink, size, etc) from other display information.

The M&C function shall allow the M&C specialist to reset existing aural and visual alarms with a single action.

After executing a command to disable alarm/normal detection, the M&C function shall provide a command response for monitored parameters with the condition "status disabled"

System Analysis Recording (SAR) – The System Analysis and Recording function provides the ability to monitor system operation, record the monitored data, and play it back at a later time for analysis. SAR data is used for incident and accident analysis, performance monitoring and problem diagnosis. The following table presents potential SAR Functional requirements.

Potential System Analysis Recording Functional Requirements

The system shall provide a System Analysis and Recording (SAR) function. (3.7.1.D) (~18800)

The SAR function shall record significant system events. (3.7.1.A) (~17890)

The SAR function shall record significant performance data. (3.7.1.A) (~17890)

The SAR function shall record significant system resource utilization. . (3.7.1.A) (~17890)

The SAR function shall record selected data while performing all system functions. (3.7.1.A) (~17890-17900)

The SAR function shall record selected system data, including system error logs, for off-line reduction and analysis of system problems and performance. (3.7.1.A) (~17890-17900)

The SAR function shall periodically record the selected data when errors/abnormal conditions are detected. (3.7.1.A) (~17890-17900)

The SAR function shall automatically dump selected memory areas when errors/abnormal conditions are detected. (3.7.1.A) (~17890-17900)

The SAR function shall record every (TBD) seconds internal state information when errors/abnormal conditions are detected. (3.7.1.A) (~17890-17900)

The data items and the conditions under which they will be recorded by the SAR function shall be determined by adaptation.

The data items and the conditions under which they will be recorded by the SAR function shall determined by M&C commands.

The SAR function shall record all system recordings on a removable storage media at a single location for a minimum of (TBD) hours without specialist intervention. . (3.7.1.A.3) (~18290)

The SAR function shall support continuous recording of system data while transitioning from one unit of recording media to another. (3.7.1.A.3) (~18290)

The SAR function shall record identifying information, including date and time, on each unit of recording media. [Include individual requirements for each that applies]. (3.7.1.A) (~17890-17900)

- Site identity
- Program version number
- Adaptation identity
- Data start/end date and time

The SAR function shall record changes in resource monitoring parameters. (3.7.1.A) (~17890-17900)

The SAR function shall record changes in recording selection parameters. (3.7.1.A) (~17890-17900)

The SAR function system shall provide off-line data reduction of recorded system data for analysis of the system's technical and operational performance.

Startup/Restart – The Startup/Restart function is one of the most critical system functions and has a significant impact on the ability of the system to meet its RMA requirements, especially for software intensive systems. The following table presents potential Startup/Restart Functional requirements.

Potential System Startup/Restart Functional Requirements

The system shall have the capability to re-establish communications and reconstitute its databases as necessary following a startup/restart.

Upon startup or restart, the system shall re-establish communications with all interfaces.

The system shall restart from a power on condition in TBD seconds. (3.8.1.D) (~19070-19090)

Software Loading and Cutover is a set of functions associated with the transfer, loading and cutover of software to the system. Cutover could be to a new release or a prior release. The following table presents potential Software Loading and Cutover Functional requirements.

Potential Software Loading and Cutover Functional Requirements

The system shall support the following tasks with no disruption to or degradation of on-going system operations or performance, except during firmware upgrades.

- *Loading of data*
- *System software*
- *Operating systems*
- *Downloadable firmware*
- *System adaptation data*

The system shall store [TBD] complete versions of application software and associated adaptation data in each system processor.

The system shall store [TBD] levels of operating system software in each system processor.

When software is loading into a processor, positive verification shall be performed to confirm that all software is loaded without corruptions, with the results reported to the M&C function. (3.7.1.A.1.c) (~18070-18080)

[If applicable.] Under the control of the M&C function and upon M&C command, the system shall cutover system processors on the non-operational data path to a previously loaded version of the software and adaptation data with no effect on the operational data path.

[If applicable.] Under the control of the M&C function and upon M&C command, shall test and evaluate software and associated adaptation versions on the non-operational data path of the system with no effect on the operational data path portion of the system.

[If applicable.] Under the control of the M&C function and upon M&C command, shall analyze performance on the non-operational data path of the system with no effect on the operational portion of the system.

[If applicable.] Under the control of the M&C function and upon M&C command, shall perform problem analysis on the non-operational data path of the system with no effect on the operational data path of the system.

Under the control of the M&C function and upon M&C command, the system shall perform system level, end-to-end tests.

The system shall perform all system level, end-to-end tests with no degradation of on-going operations or system performance.

Certification – Certification is an inherently human process of analyzing available data to determine if the system is worthy of performing its intended function. One element of data is often the results of a certification function that is designed to exercise end-to-end system functionality using known data and

predicable results. Successful completion of the certification function is one element of data used by the Specialist to determine the system is worthy of certification. Some systems employ a background diagnostic or verification process to provide evidence of continued system certifiability. The following table presents potential Certification Functional requirements.

Potential Certification Functional Requirements

Prior to allowing an off-line LRU to be configured as part of the on-line operational system, the M&C function shall automatically initiate comprehensive tests/diagnostics on that off-line LRU (to the extent possible for that LRU), and report the results to the M&C position. (3.7.1.B) (~18320)

The M&C function shall automatically perform real-time, on-line, periodic tests without interruption or degradation to operations on all LRUs, and reporting any out-of-tolerance results to the M&C position. (3.7.1.B) (~18320)

The M&C function shall, upon M&C command, modify the frequency of the background verification tests, from a minimum frequency of once every (TBD) hours to a maximum frequency of once every (TBD) minutes. (3.7.1.B) (~18320)

The M&C function shall, upon M&C command, initiate the background verification test for a specified LRU, and receive a hard copy printout of the test results. (3.7.1.B) (~18320)

The M&C function shall provide on-line system certification of the entire system without interruption or degradation to operations. (3.7.1.C.1.c) (~18450)

The M&C function shall, upon M&C command, manually initiate on-line certification of the system. (3.7.1.C.1.c) (~18450)

Transition – Transition is a set of requirements associated with providing functionality required to support the transition to or upgraded to new systems. The following table presents potential Transition Functional requirements.

Potential Transition Functional Requirements

The M&C function shall inhibit inputs to the system when the system is in a test/monitor mode to prevent inadvertent interference with ATC operations. (3.7.3)

The M&C function shall concurrently perform system-level testing and shadow mode testing and training at system positions without affecting on-going ATC operations. (3.7.2.A) (3.7.3)

The M&C function shall reconstitute displayed data such that all outputs needed for operations are available at the selected controlling position equipment.

Maintenance support is a collection of requirements associated with performing preventative and corrective maintenance of equipment and software. The following table presents potential Maintenance

Support Functional requirements.

Potential Maintenance Support Functional Requirements

The M&C function shall control the facilities, equipment, and systems necessary to perform preventive maintenance activities including adjustment, diagnosis, replacement, repair, reconditioning, and recertification. (3.7.1.C) (~18360)

The M&C function shall control the facilities, equipment, and systems necessary to perform corrective maintenance activities including adjustment, diagnosis, replacement, repair, reconditioning, and recertification. (3.7.1.C) (~18370)

The system shall provide test circuitry and analysis capabilities to allow diagnosis of the cause of a system/equipment failure, isolation of the fault, and operational checkout. (3.7.1.C.2) (~18370)

Test Support Functions – Test support is a collection of requirements associated with supporting system testing before, during and after installation of the system. The following table presents potential Test Support Functional requirements.

Potential Test Support Functional Requirements

The system shall provide test sets, test drivers, scenarios, simulators and other test support items required to provide a realistic test environment. (3.7.3)

The system shall record, reduce and analyze the test data. (3.7.3)

Training support is a collection of requirements associated with supporting training of system specialists. The following table presents potential M&C Training requirements.

Potential M&C Training Requirements

The system shall perform training operations concurrently with ongoing ATC operations, with no impact to operations. (3.7.2.A)

The M&C function shall configure system resources to support Air Traffic operational training in a simulated training environment. (3.7.2.A)

Upon M&C command, the M&C function shall initiate Air Traffic operational training in a simulated training environment (3.7.2.A)

Upon M&C command, the M&C function shall terminate Air Traffic operational training in a simulated training environment. (3.7.2.A)

Operational software shall be used in training exercises. (3.7.2.A)

Appendix B RELIABILITY AND AVAILABILITY TABLES FOR REPAIRABLE REDUNDANT SYSTEMS

B.1 Availability Table

Error! Reference source not found. illustrates the improvement in availability achieved by adding a redundant element. The table can be used to assist in the evaluation of inherent availability models of redundant systems.

TABLE B - 1 Combinatorial Availability for a "Two Needing One" Redundant Configuration

Element Availability	System Availability for N = 2, R = 1
0.99	0.9999
995	0.999975
0.999	0.999999
0.9995	0.99999975
0.9999	0.99999999
0.99995	1.00000000
0.99999	1.00000000
0.999995	1.00000000
0.999999	1.00000000
0.9999995	1.00000000
0.9999999	1.00000000

B.2 Mean Time between Failure (MTBF) Graphs

The graphs shown in **Error! Reference source not found.**, **Error! Reference source not found.** , and **Error! Reference source not found.** illustrate the reliability improvement achieved with a dual redundant configuration. The X-axis represents the MTBF of a single element, and Y axis represents the MTBF of the redundant configuration. The system reliability for repairable redundant systems is also affected by the time to return failed elements to service. The separate curves on each graph represent different values of MTTR.

The three graphs are based on different ranges of reliability of the individual elements comprising the redundant configuration. The charts were computed using the Einhorn equations presented in Appendix C.

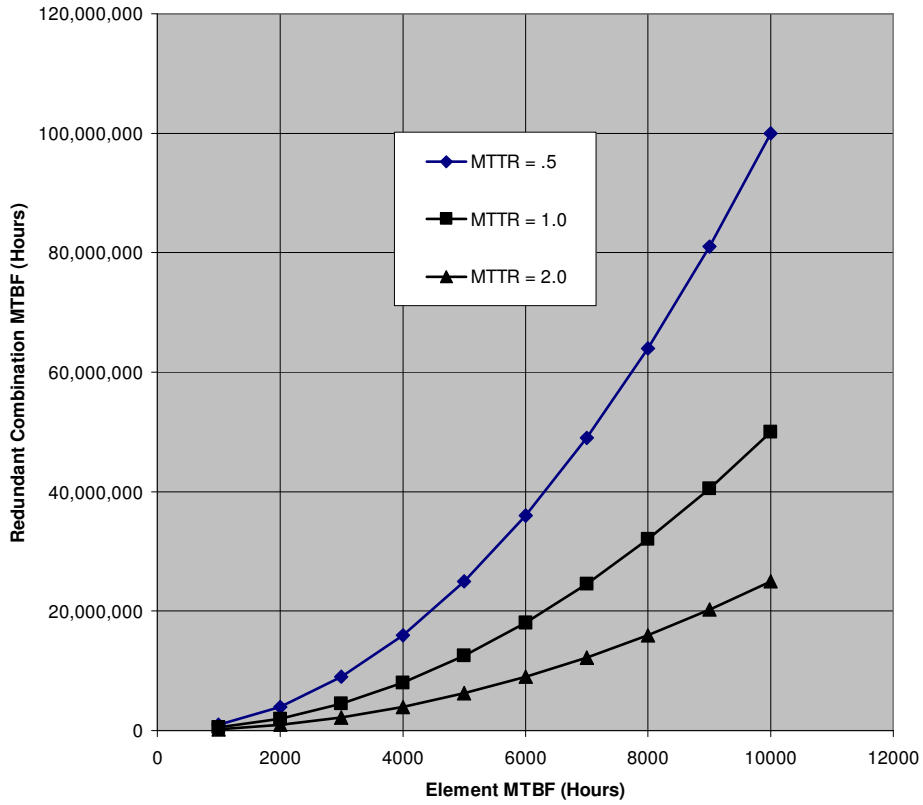

FIGURE B - 1: Mean Time between Failure for a "Two Needing One" Redundant Combination (a)

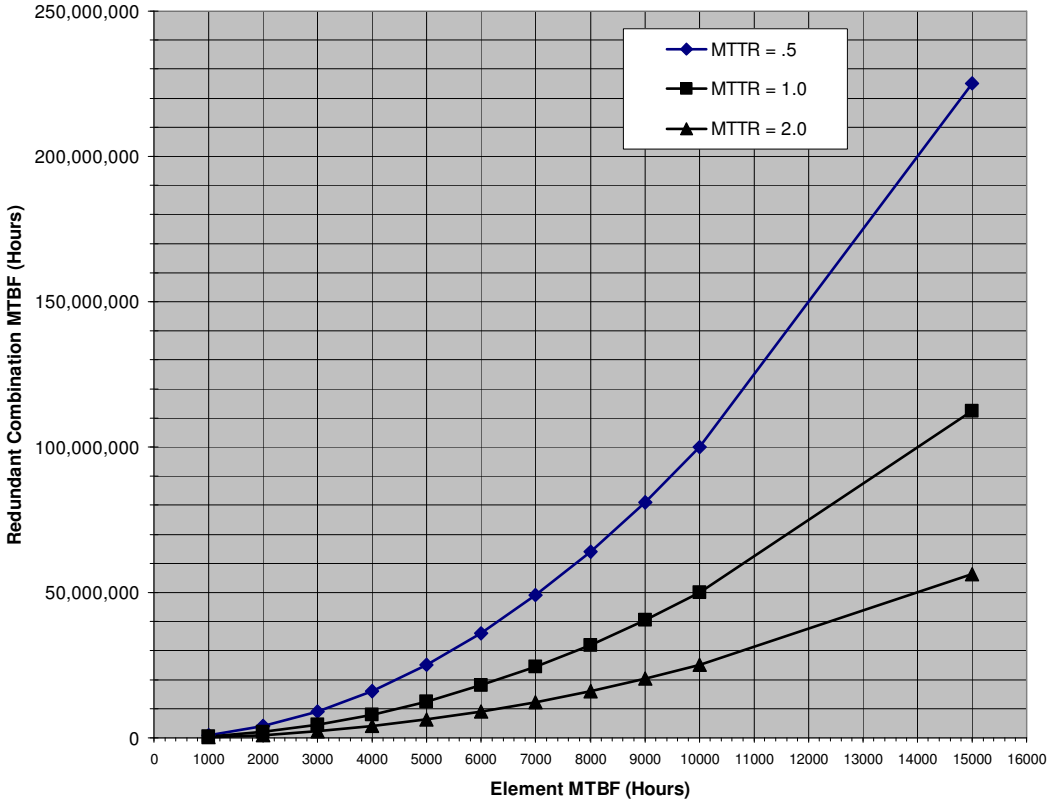

FIGURE B - 2: Mean Time between Failure for a "Two Needing One" Redundant Combination (b)

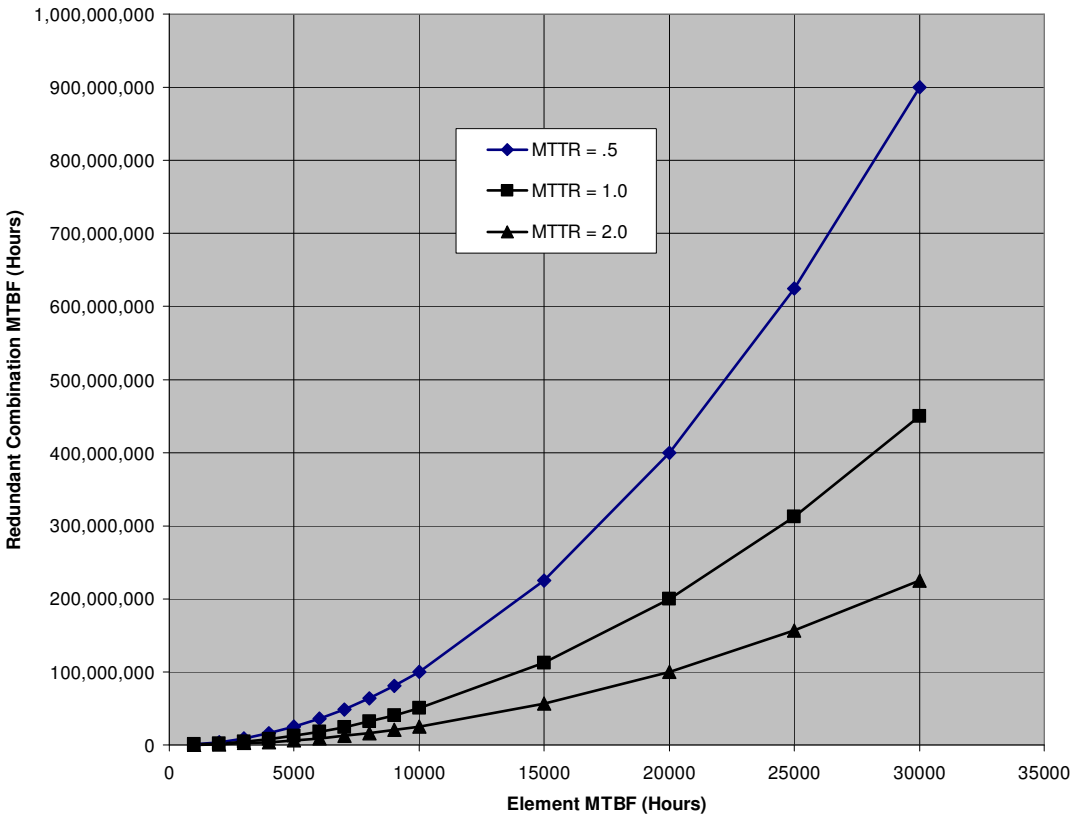

FIGURE B - 3: Mean Time between Failure for a "Two Needing One" Redundant Combination (c)

Appendix C STATISTICAL METHODS AND LIMITATIONS

C.1 Reliability Modeling and Prediction

The statistical basis for reliability modeling was originally developed in the 1950's when electronic equipment was fabricated with discreet components such as capacitors, resistors, and transistors. The overall reliability of electronic equipment is related to the numbers and failure rates of the individual components used in the equipment. Two fundamental assumptions form the basis for conventional parts count reliability models:

- The failure rates of components are assumed to be constant. (After a short initial burn-in interval and before end-of-life wear out—the "bathtub curve.")
- The failures of individual components occur independently of one another.

The constant failure rate assumption allows the use of an exponential distribution to describe the distribution of time to failure, so that the probability a component will survive for time *t* is given by

$$R = e^{-\lambda t} \qquad \text{[C-1]}$$

(Where *R* is the survival probability, λ is the constant failure rate, and *t* is the time.)

The assumption of independent failures, means if that the failure of one component does not affect the probability of failure of another component, then the probability of all components surviving is the product of the individual survival probabilities.

$$R_T = R_1 * R_2 * \cdots\cdots R_n \qquad \text{[C-2]}$$

Because of the exponential distribution of failures, the total failure rate is simply the sum of the individual failure rates and the total reliability is

$$R_T = e^{-\lambda_T t} \qquad \text{[C-3]}$$

Where λ_T is given by

$$\lambda_T = \sum_{i=1}^{n} \lambda_i \qquad \text{[C-4]}$$

The equation for predicting the equipment failure rate using the parts count method is given by MIL-HDBK-217 as

$$\lambda_{Equip} = \sum_{i=1}^{n} N_i (\lambda_G \Pi_Q) \qquad \text{[C-5]}$$

Where

λ_{EQUIP} = Total equipment failure rate (failures/10^6 hours)

λ_G = Generic failure rate for the i^{th} generic part (failures/10^6 hours)

Π_Q = Quality factor for the i^{th} generic part

N_I = Quantity of i^{th} generic part

N = Number of different generic part categories in the equipment

This reliability prediction technique worked reasonably well for simple "black box" electronic equipment. However, the introduction of fault-tolerant redundant computer systems created a need for more complex modeling techniques that are discussed in Section C.4

C.2 Maintainability

Maintainability is defined in MIL-STD-721 as "The measure of the ability of an item to be retained in or restored to specified condition when maintenance is performed by personnel having specified skill levels, using prescribed procedures and resources, at each prescribed level of maintenance and repair."

Maintainability prediction methods depend primarily on two basic parameters, the failure rates of components at the level of maintenance actions, and the repair or replacement time for the components. Historically, maintainability predictions for electronic equipment involved a detailed examination of the components' failure rates and the measured time required for diagnosing and replacing or repairing each of the failed components. A statistical model combined the failure rates and repair times of the equipment's components to determine an overall Mean Time to Repair (MTTR) or Mean Down Time (MDT).

Maintainability was a design characteristic of the equipment. Repair times were affected by the quality of built in test equipment (BITE), diagnostic tools, and the ease of access, removal, and replacement of failed components.

With the advent of redundant, fault-tolerant systems, in which restoration of service is performed by automatic switchover and corrective maintenance is performed off-line, maintainability is not as significant as it once was. In addition, the move to utilizing more commercial off the shelf (COTS) equipment that is simply removed and replaced has made the traditional maintainability calculations as expressed in MIL-HDBK-472 less relevant.

C.3 Availability

Availability is defined in MIL-STD-721 as a measure of the degree to which an item is in an operable and Committable State at the start of a mission when the mission is called for at an unknown (random) time. As such, availability is the probability that the system will be available when needed, and the availabilities for independent subsystems can be combined by simply multiplying the availabilities. (Availability is also used to express the percentage of units that may be available at the start of a mission, e.g. how many aircraft in a squadron will be available at the start of a mission.)

Availability is measured in the field by subtracting the down time from the total elapsed time to obtain the time that the system was operational and dividing this time by the total elapsed time. Operational availability includes all downtime. Other availability measures have been defined that exclude various categories of down time such as those caused by administrative delays and logistics supply problems. The purpose of these other measures of availability is to develop metrics that more accurately reflect the

characteristics of the system itself removing downtime that is attributable to deficiencies in the human administration of the system.

Availability is usually not predicted directly, but is usually derived from both the failure and repair characteristics of the equipment. Availability is expressed as

$$A = \frac{MTBF}{MTBF + MTTR} \qquad \text{[C-6]}$$

Where MTBF is the Mean Time between Failures and MTTR is the mean time to repair, or equivalently

$$A = \frac{MUT}{MUT + MDT} \qquad \text{[C-7]}$$

(Where MUT is the Mean Up Time and MDT is the Mean Down Time.)

As discussed earlier, availability allows reliability and maintainability to be traded off. Although this practice may be acceptable for equipment where optimizing life cycle costs is the primary consideration, it is may not be appropriate for systems that provide critical services to air traffic controllers, where lengthy service interruptions may be unacceptable, regardless of how infrequently they are predicted to occur.

C.4 Modeling Repairable Redundant Systems

The increasing use of digital computers for important real-time and near-real-time operations in the 1960's created a demand for systems with much greater reliability than that which could be achieved with the current state of the art for electronic systems constructed with large numbers of discrete components. For example, the IBM 360 series computers employed in NAS Stage A had a MTBF on the order of 1000 hours. The path to higher reliability systems was to employ redundancy and automatic fault detection and recovery. The introduction of repairable redundant systems required new methods for predicting the reliability of these systems. One of the first attempts at predicting the reliability of these systems was presented in a paper by S. J. Einhorn in 1963. He developed a method for predicting the reliability of a repairable redundant system using the mean time to failure and mean time to repair for the elements of the system. He assumed that the system elements conformed to the exponential failure and repair time distributions and that the failure and repair behaviors of the elements are independent of one another. The Einhorn equation for predicting the reliability of an r out n redundant system is presented below.

$$MUT = \frac{\sum_{j=r}^{n} \binom{n}{j} U^{j} D^{n-j}}{n \binom{n-1}{r-1} U^{r-1} D^{n-r}} \qquad \text{[C-8]}$$

Where MUT is the mean UP time, n is the total number of elements in a subsystem, r is the number of elements that are required for the system to be UP, and the number of combinations of n things taken r at a time is given by

$$\binom{n}{r} = \frac{n(n-1)(n-2)\ldots\ldots(n-r+1)}{r!} = \frac{n!}{r!(n-r)!} \qquad [C\text{-}9]$$

The Einhorn method provided a relatively simple way to predict the combinatorial reliability of an "r out of n" repairable redundant configuration of identical elements. This method assumes perfect fault detection, isolation and recovery and does not account for switchover failures or allow for degraded modes of operation.

In order to incorporate these additional factors, Markov models were developed to model reliability and availability. Typically, Markov models of redundant systems assume that the overall system is organized as a set of distinct subsystems, where each subsystem is composed of identical elements and the failure of a subsystem is independent of the status of the other subsystems. In each of the subsystems, redundancy is modeled by a Markov process with the following typical assumptions:

- Failure rates and repair rates are constants.
- System crashes resulting from the inability to recover from some failures even though operational spares are available are modeled by means of "coverage" parameters. (Coverage is defined as the probability that the system can recover, given that a fault has occurred.)
- For recoverable failures, the recovery process is instantaneous if useable spares are available.
- Spare failures are detected immediately.
- As soon as a failed unit is repaired, it is assumed to be in perfect condition and is returned to the pool of spares.

Reliability analysis using Markov models follows four distinct steps:

1. Development of the state transition diagram
2. Mathematical representation (Differential equation setup)
3. Solution of the differential equations
4. Calculation of the reliability measures

An example of a general state transition diagram for a system with three states is provided by . The circles represent the possible states of the system and the arcs represent the transitions between the states. A three-state model is used to represent the behavior of a simple system with two elements, one of which must be operational for the system to be up. In State 1, both elements are operational. In State 2, one of the elements has failed, but the system is still operational. In State 3, both elements have failed and the system is down.

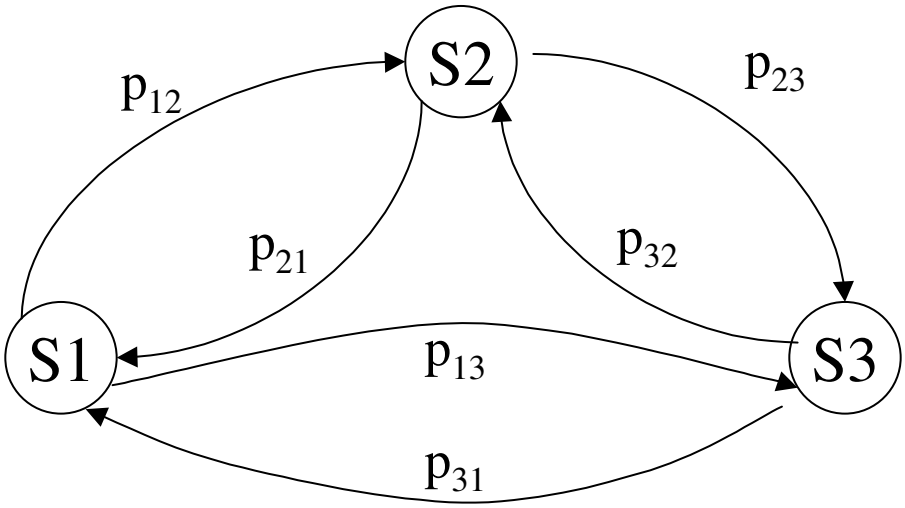

FIGURE C 1

: General State Transition Diagram for Three-State System

From the state transition diagram in a set of differential equations can be formulated as follows:

If a system is in State 1 at time t + Δt, then between time t and t + Δt, one of two events must have occurred: (1) either the system was in state 1 at time t and stayed in that state throughout the interval Δt, or (2) it was in State 2 or State 3 at time t and a transition to State 1 occurred in the interval Δt.

The probability of event 1, that the system stayed in State 1 throughout the interval Δt is equal to one minus the probability that a transition occurred from State 1 to either State 2 or State 3.

$$P(E1) = \left[1 - \left(p_{12}\Delta t + p_{13}\Delta t\right)\right]P_1(t) \qquad \text{[C-10]}$$

The probability of event 2 is given by the probability that the system was in State 2 times the probability of a transition from State 2 to State 1 in Δt, plus the probability that the system was in State 3 times the probability of a transition from State 3 to State 1 in Δt.

$$P(E2) = (p_{21}\Delta t)P_2(t) + (p_{31}\Delta t)P_3(t) \qquad \text{[C-11]}$$

Since the two events are statistically independent, the probability of being in State 1 at time t + Δt is the sum of the probabilities of the two events

$$P_1(t+\Delta t) = \left[1 - (p_{12} + p_{13})\Delta t\right]P_1(t) + \left[p_{21}P_2(t) + p_{31}P_3(t)\right]\Delta t \qquad \text{[C-12]}$$

Rearranging the terms in Equation [C-12] and letting Δt approach zero, yields the following differential equation

$$\frac{d}{dt}P_1(t) = -(p_{12} + p_{13})P_1(t) + p_{21}P_2(t) + p_{31}P_3(t) \qquad \text{[C-13]}$$

Similarly, the equations for the other two states are

$$\frac{d}{dt}P_2(t) = p_{12}P_1(t) - (p_{21} + p_{23})P_2(t) + p_{32}P_3(t) \quad \text{[C-14]}$$

$$\frac{d}{dt}P_3(t) = p_{13}P_1(t) + p_{23}P_2(t) - (p_{31} + p_{32})P_3(t) \quad \text{[C-15]}$$

Equations [C-13], [C-14], and [C-15] can be written in matrix form as

$$\frac{d}{dt}\underline{P}(t) = A\underline{P(t)} \quad \text{[C-16]}$$

Or

$$\frac{d}{dt}\begin{bmatrix} P_1(t) \\ P_2(t) \\ P_3(t) \end{bmatrix} = \begin{bmatrix} -(p_{12} + p_{13}) & p_{21} & p_{31} \\ p_{12} & -(p_{21} + p_{23}) & p_{32} \\ p_{13} & p_{23} & -(p_{31} + p_{32}) \end{bmatrix} \begin{bmatrix} P_1(t) \\ P_2(t) \\ P_3(t) \end{bmatrix} \quad \text{[C-17]}$$

Where A represents the transition probability matrix (TPM) for the state diagram and the elements of the matrix represent the transition rates between states. In a reliability or availability model, these rates are determined primarily by the failure and repair rates of the system elements.

Typically, the State Transition Diagram will not include all of the possible state transitions shown in **Error! Reference source not found.**. For example, reliability models generally do not include any transitions out of the failed state, while availability models will add a repair rate out of the failed state corresponding to the time to restore the system to full operation following a total failure. Other transitions between the states in **Error! Reference source not found.** may not be possible in the particular system being modeled.

Error! Reference source not found. presents a simplified transition diagram for a system with two elements, one of which is required for full operation. S1 is the state when both elements are up, S2 is the state when one element is up and the other failed, but the system is still operational. S3 is the failed state when neither of the two elements is operational and the system is down. The only transitions between states are a result of the failure or repair of an element. The transition probabilities for the paths in the general model in **Error! Reference source not found.** that are not shown in **Error! Reference source not found.** are set to zero. Since this is a reliability model that reflects the time to failure of the system, there are no transitions out of the failed state, S3. It is considered in Markov terminology to be an *absorbing state*. Once the system has run out of spares and failed, it stays in the failed state indefinitely. This simplified model addresses only the combinatorial probability of encountering a second failure before the first failure has been repaired, i.e. exhausting spares. It does not consider failures of automatic switchover mechanisms or address other factors such as degraded states, undetected spare failures, etc.

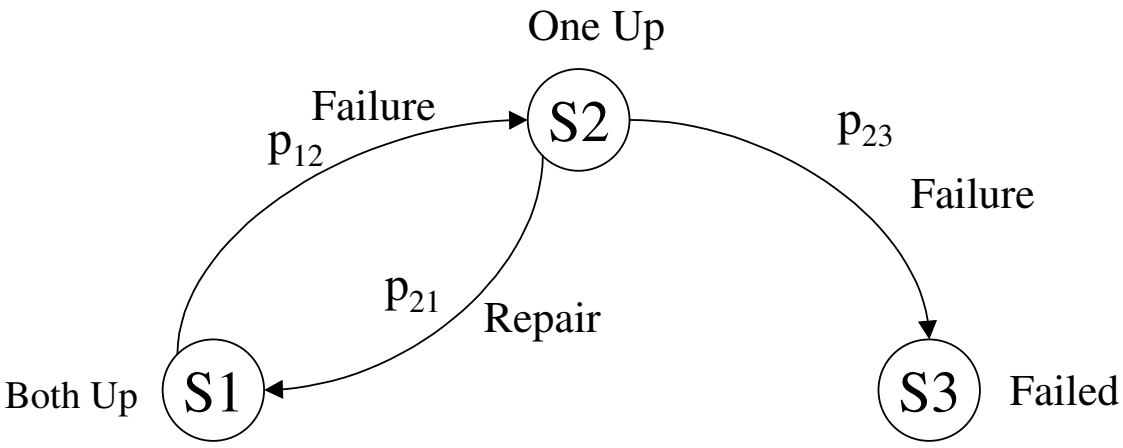

FIGURE C - 1: Simplified Transition Diagram

This simple example can be used to illustrate how the differential equations can be solved. Suppose that each element has a failure rate of 0.001 failures/hour (1000 hour MTBF) and a repair rate of 2 repairs/hour (0.5 hours MTTR). The transition probabilities are then

p_{12} = .002 (because there are two elements each having a .001 failure rate)

p_{23} = .001 (because there is only one element left to fail)

p_{21} = 2

All of the other transition probabilities are zero

Thus the transition probability matrix of Equation [C-17] becomes

$$A = \begin{bmatrix} -.002 & 2 & 0 \\ .002 & 2.001 & 0 \\ 0 & .001 & 0 \end{bmatrix} \qquad [C\text{-}18]$$

Since, for reliability prediction, the reliability is expressed by the probability of being in one of the two "UP" states S1 or S2, the TPM can be further simplified to

$$A = \begin{bmatrix} -.002 & 2 \\ .002 & -2.001 \end{bmatrix} \qquad [C\text{-}19]$$

Equations [C-13] and [C-14] then become

$$\frac{d}{dt}P_1(t) = -.002P_1(t) + 2P_2(t) \qquad [C\text{-}20]$$

$$\frac{d}{dt}P_2(t) = .002P_1(t) - 2.001P_2(t) \qquad [C\text{-}21]$$

Taking the Laplace transform of these equations yields

$$\begin{aligned} sP_1(s) - P_1(0+) &= -.002P_1(s) + 2P_2(s) \\ sP_2(s) - P_2(0+) &= .002P_1(s) + 2.001P_2(s) \end{aligned} \qquad [C\text{-}22]$$

Rearranging terms and substituting the initial conditions for P_1 (0+) and P_2 (0+)

$$\begin{aligned} (s + .002)P_1(s) - 2P_2(s) &= 1 \\ .002P_1(s) + (s + 2.001)P_2(s) &= 0 \end{aligned} \qquad (C\text{-}23)$$

The equations in (C-23) can be solved using Cramer's rule as

$$P_1(s) = \frac{\begin{vmatrix} 1 & -2 \\ 0 & (s+2.001) \end{vmatrix}}{\begin{vmatrix} (s+.002) & -2 \\ -.002 & (s+2.001) \end{vmatrix}} = \frac{(s+2.001)}{s^2 + 2.003s + 2\times10^{-6}}$$

$$P_2(s) = \frac{\begin{vmatrix} (s+.002 & 1 \\ -.002 & 0 \end{vmatrix}}{\begin{vmatrix} (s+.002) & -2 \\ -.002 & (s+2.001) \end{vmatrix}} = \frac{.002}{s^2 + 2.003s + 2\times10^{-6}} \qquad [C\text{-}24]$$

The factors of the denominator are (s + 2.003) and (s + 10^{-6}). Expanding Equations [C-24] by partial fractions yields

$$\begin{aligned} P_1(s) &= \frac{.000998503}{(s+2.003)} + .\frac{.999001497}{(s+10^{-6})} \\ P_2(s) &= \frac{-.000998503}{(s+2.003)} + \frac{.000998503}{(s+10^{-6})} \end{aligned} \qquad [C\text{-}25]$$

Since the reliability is given by the sum of the probabilities of being in State 1 or 2, then the reliability is

$$R(s) = \frac{1}{(S+10^{-6})} \qquad [C\text{-}26]$$

and taking the inverse Laplace transformation,

$$R(t) = e^{-10^{-6}t} \qquad \text{[C-27]}$$

This indicates that the reliability is equal to 1.0 at t=0 and decays exponentially as t increases. The system mean time between failures (MTBF) is given by the reciprocal of the failure rate in the exponent or one million hours. This is the same result that is obtained by using the Einhorn equations [C-8]

In this simple example, there is virtually no difference between a Markov model and the Einhorn equations. Note that Markov models can be extended almost indefinitely to include additional system states and transitions between states. For example, our simple reliability model in **Error! Reference source not found.** can be extended to include the effects of failure to detect an element failure or successfully switch to a spare element by adding the transition path shown in FIGURE C - 2:

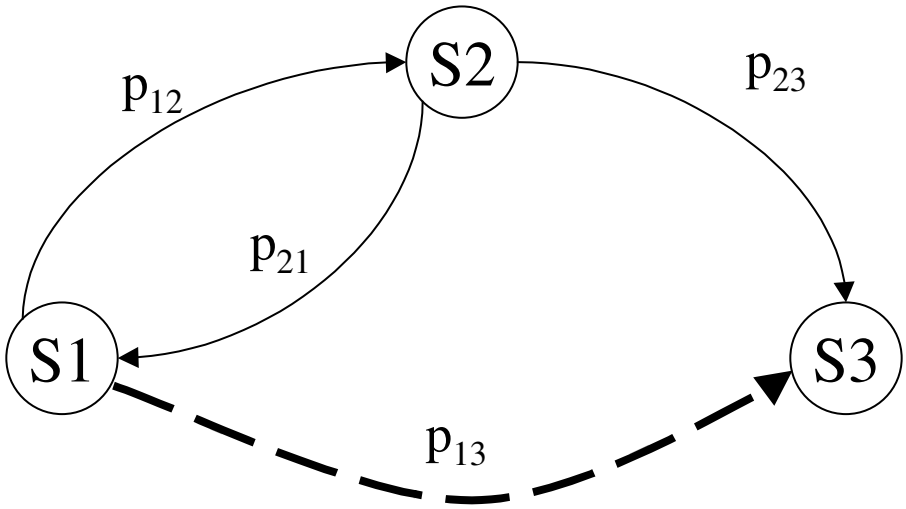

FIGURE C - 2: Coverage Failure

The transition path p_{13} represents a crash failure of the automatic fault detection and recovery mechanisms. The transition rate from the full up state to the failed state is dependent on the failure rate of the system elements and the value of the coverage parameter, C. Coverage is a dimensionless parameter between zero and one that represents probability that recovery from a failure is successful, given that a failure has occurred. The value of p_{13} is given by

$$p_{13} = 2\lambda(1-C) \qquad \text{[C-28]}$$

If the coverage is perfect with C equal to one, then the transition probability from S1 to S3 is zero and FIGURE C - 2: becomes equivalent to **Error! Reference source not found.**. If C is equal to zero, then the automatic recovery mechanisms never work and the system will fail whenever either of the two elements fail, (assuming that the automatic recovery mechanisms are invoked whenever a failure occurs anywhere in the system, a common practice in systems employing standby redundancy).

If a model of availability instead of reliability is desired, it will be necessary to add a recovery path from the failed state as in the availability model shown in **Error! Reference source not found.**.

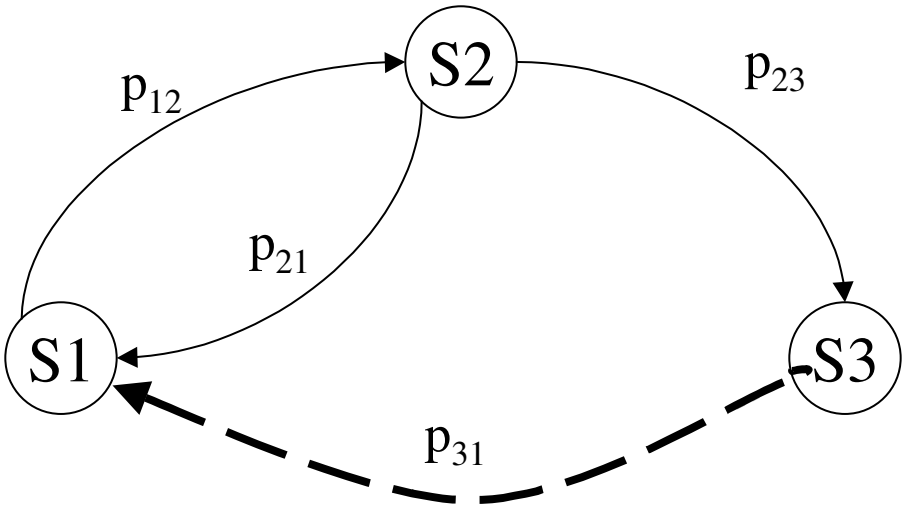

FIGURE C - 3: Availability Model

The transition from the failed state to the full up state, p_{31} is twice the repair rate for a single element if the capability exists to repair and restore both failed elements simultaneously.

The preceding examples illustrate some very simple Markov models. The number of states and transition paths can be extended indefinitely to include a wide variety of system nuances such as degraded modes of operation, and undetected failures of spare elements. However the number of elements in the transition probability matrix increases as the square of the number of states making the hand calculations illustrated above a practical impossibility. Although the solution mathematics and methodology are the same, the sheer number of arithmetic manipulations required makes the solution of the equations a time-consuming and error-prone process. For this reason Markov modeling is usually performed using computer tools. Many of these tools can automatically construct the transition probability matrix (TPM) from the input parameters, solve the differential equations using numerical methods, and then calculate a variety of RMA measures from the resulting state probabilities.

C.5 Availability Allocation

A typical reliability block diagram consists of a number of independent subsystems in series. Since the availability of each subsystem is assumed to be independent of the other subsystems, the total availability of the series string is given by

$$A_{Total} = A_1 \times A_2 \cdots\cdots A_n \qquad \text{[C-29]}$$

The most straightforward method of allocating availability is to allocate the availability equally among all of the subsystems in the reliability block diagram. The allocated availability of each element in the reliability block diagram is then given by

$$A_{Subsystem} = (A_{Total})^{\frac{1}{n}} \qquad \text{[C-30]}$$

A simpler approximation can be derived by rewriting the availability equation using the expression

$$A = (1 - \overline{A}) \qquad \text{[C-31]}$$

where $\overline{A}$ represents the unavailability. Rewriting equation [C-29]

$$A_{Total} = (1-\overline{A}_1)\times(1-\overline{A}_2)\ldots(1-\overline{A}_n) \qquad \text{[C-32]}$$

Multiplying terms and discarding higher order unavailability products yields the following approximation

$$A_{Total} = (1-n\overline{A}_{Subsystem}) \qquad \text{[C-33]}$$

or by rearranging terms

$$\overline{A}_{Subsystem} = \frac{1}{n}\times\overline{A}_{Total} \qquad \text{[C-34]}$$

The approximation given by Equation [C-34] allows the availability allocation to be performed by simple division instead of calculating the n^{th} root of the total availability as in Equation [C-30].

Thus to allocate availability equally across n independent subsystems, it is only necessary to divide the total unavailability by the number of subsystems in the series string to determine the allocated unavailability for each subsystem. The allocated availability for each subsystem then is simply

$$A_{Subsystem} = \left[1-\frac{\overline{A}_{Total}}{n}\right] \qquad \text{[C-35]}$$

At this point, it is instructive to reflect on where all of this mathematics is leading. Looking at equation [C-35] if n = 10, the allocated availability required for each subsystem in the string will be an order of magnitude greater than the total availability of the Service Thread. This relationship holds for any value of total availability. For a Service Thread with ten subsystems, the allocated availability for each subsystem in a Service Thread will always be one "nine" greater than the number of "nines" required for the total availability of the Service Thread. Since none of the current threads has ten subsystems, all availability allocations will be *less* than an order of magnitude greater than the total availability required by the Service Thread. By simply requiring the availability of any system in a thread to be an order of magnitude greater than the end-to-end availability of the thread, the end-to-end availability of the thread will be ensured unless there are more than 10 systems in the thread. This convention eliminates the requirement to perform a mathematical allocation and eliminates the issue of whether the NAS-Level availability should be equally allocated across all systems in the thread. Mathematical allocations also contribute to the illusion of false precision. It is likely that allocations will be rounded up to an even number of "nines" anyway. The risk, of course, of requiring that systems have availability an order of magnitude greater that the threads they support is that the system availability requirement is greater than absolutely necessary, and could conceivably cause systems to be more costly.

This should not be a problem for two reasons. First, as discussed in Section 7.1.1, this process only applies to information systems. Other methods are proposed for remote and distributed elements and facility infrastructure systems. Secondly, a system designer is only required to show that the architecture's *inherent* availability meets the allocated requirement. The primary decision that needs to be made is whether the system needs to employ redundancy and automatic fault detection and recovery. With no redundancy, an inherent availability of three to four "nines" is achievable. With minimum redundancy, the inherent availability will have a quantum increase to six to eight "nines." Therefore, allocated availabilities in the range of four to five "nines" will not drive the design. Any availability in this range

will require redundancy and automatic fault detection and recovery that should easily exceed the allocated requirement of five “nines.”

C.6 Modeling and Allocation Issues

RMA Models are a key factor in the process of the allocating system of allocating NAS-Level requirements to the systems that are procured to supply the services and capabilities defined by the NAS requirements. At this point, although the mathematics used in RMA modeling may appear elegant at first glance, it is appropriate to reflect upon the limitations of the statistical techniques used to predict the RMA characteristics of modern information systems.

Although the mathematics used in RMA models is becoming increasing sophisticated, there is a danger in placing two much confidence in these models. This is especially important when the results can be obtained by entering a few parameters into a computer tool without a clear understanding of the assumptions embedded in the tool and the sensitivity of the model results to variations in the input parameters. One of the most sensitive parameters in the model of a fault-tolerant system is the coverage parameter. The calculated system reliability or availability is almost entirely dependent on the value chosen for this parameter. Minor changes in the value of coverage cause wide variations in the calculated results. Unfortunately, it is virtually impossible to predict the coverage with enough accuracy to be useful.

This raises a more fundamental issue with respect to RMA modeling and verification. The theoretical basis of RMA modeling rests on the assumptions of constant failure rates and the statistical independence of physical failures of hardware components. The model represents a “steady state” representation of a straightforward physical situation.

Physical failures are no longer the dominate factor in system reliability and availability predictions. Latent undiscovered design defects created by human beings in the development of the system predominate as causes of failures. Although attempts have been made to incorporate these effects into the models, some fundamental problems remain. First, conventional reliability modeling used an empirical database on component failure history to predict the reliability of systems constructed with these components. Historical data has been collected on software fault density (Number of faults/ksloc). It is difficult, however, to translate this data into meaningful predictions of the system behavior without knowing how the faults affect the system behavior and how often the code containing a fault will be executed.

Secondly, the reliability of a computer system is not fundamentally a steady state situation, but a reliability growth process of finding and fixing latent design defects. Although some have argued that software reliability eventually reaches a steady state in which fixing problems introduces an equal number of new problems, there is no practical way to relate the latent fault density of the software (i.e. bugs/ksloc) to its run-time failure rate (i.e. failures/hour). There have been some academic attempts to relate the fault density to the run-time performance of the software, the usefulness of such predictions is questionable. They are of little value in acquiring and accepting new systems, and uncertainty concerning the frequency of software upgrades and modifications makes the prediction of steady state software reliability of fielded systems problematic.

Because of the questionable realism and accuracy of RMA predictions from sophisticated RMA models, it is neither necessary nor desirable to make the allocation of NAS requirements to systems unnecessarily complicated. It should not require a Ph.D. in statistics or a contract with a consulting firm to perform them. Accordingly, throughout the development of the allocation process, the objective is to make the process understandable, straightforward, and simple so that a journeyman engineer can perform the allocations.

Appendix D FORMAL RELIABILITY DEMONSTRATION TEST PARAMETERS

MIL-STD-781 defines the procedures and provides a number of suggested test plans for conducting reliability demonstration tests. The statistics and equations defining the characteristics of test plans are also presented. For a detailed description formal reliability testing, the reader is referred to MIL-STD-781.

This appendix summarizes the fundamental statistical limitations underlying the testing issues introduced in Section 5.2.2.2.

Error! Reference source not found.illustrates what is known in statistics as an Operating Characteristic (OC) Curve. The OC curve presents the probability of accepting a system versus multiples of the test MTBF. An ideal reliability qualification test would have the characteristics of the heavy dashed line. Unfortunately, basing the decision of whether to accept or reject a system on the basis of a limited sample collected during a test of finite duration has an OC more like the other curve that represents a test scenario from MIL-STD-781 in which the system is tested for 4.3 times the required MTBF. The system is accepted if it incurs two or fewer failures during the test period and rejected if it incurs 3 or more failures during the test period.

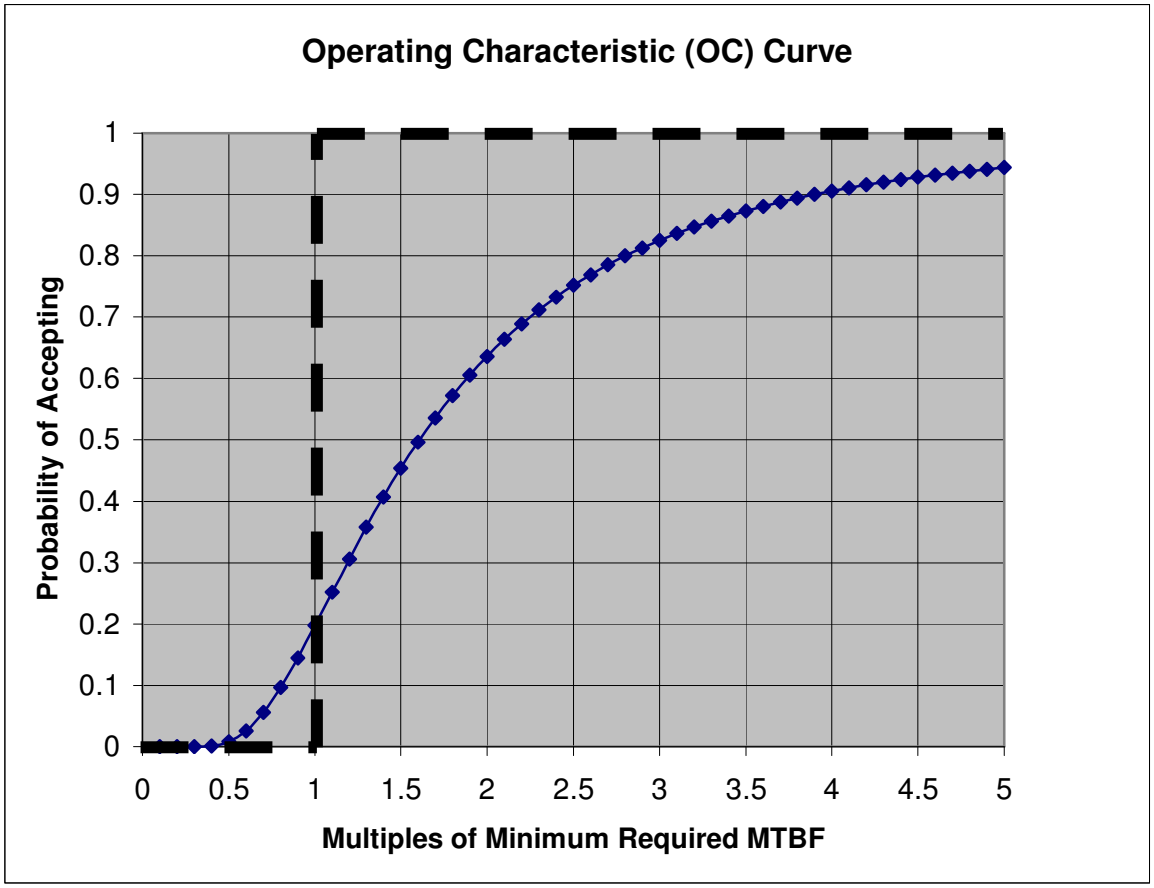

FIGURE D - 1: Operating Characteristic Curves

An explanation of the important points on an OC curve is illustrated in **Error! Reference source not found.**. This curve represents the fixed length test described in the preceding paragraph. There are two types of incorrect decisions that can occur when an acceptance decision is based on a limited data sample. The Type I error occurs when a "good" system whose true reliability meets or exceeds the requirements fails the test. The probability of occurrence of the Type I error is given by α and is known as the producer's risk. The Type II error occurs when the test passes a "bad" system whose true MTBF is below the minimum acceptable requirement. The probability of the Type II error is given by β and is known as the consumer's risk.

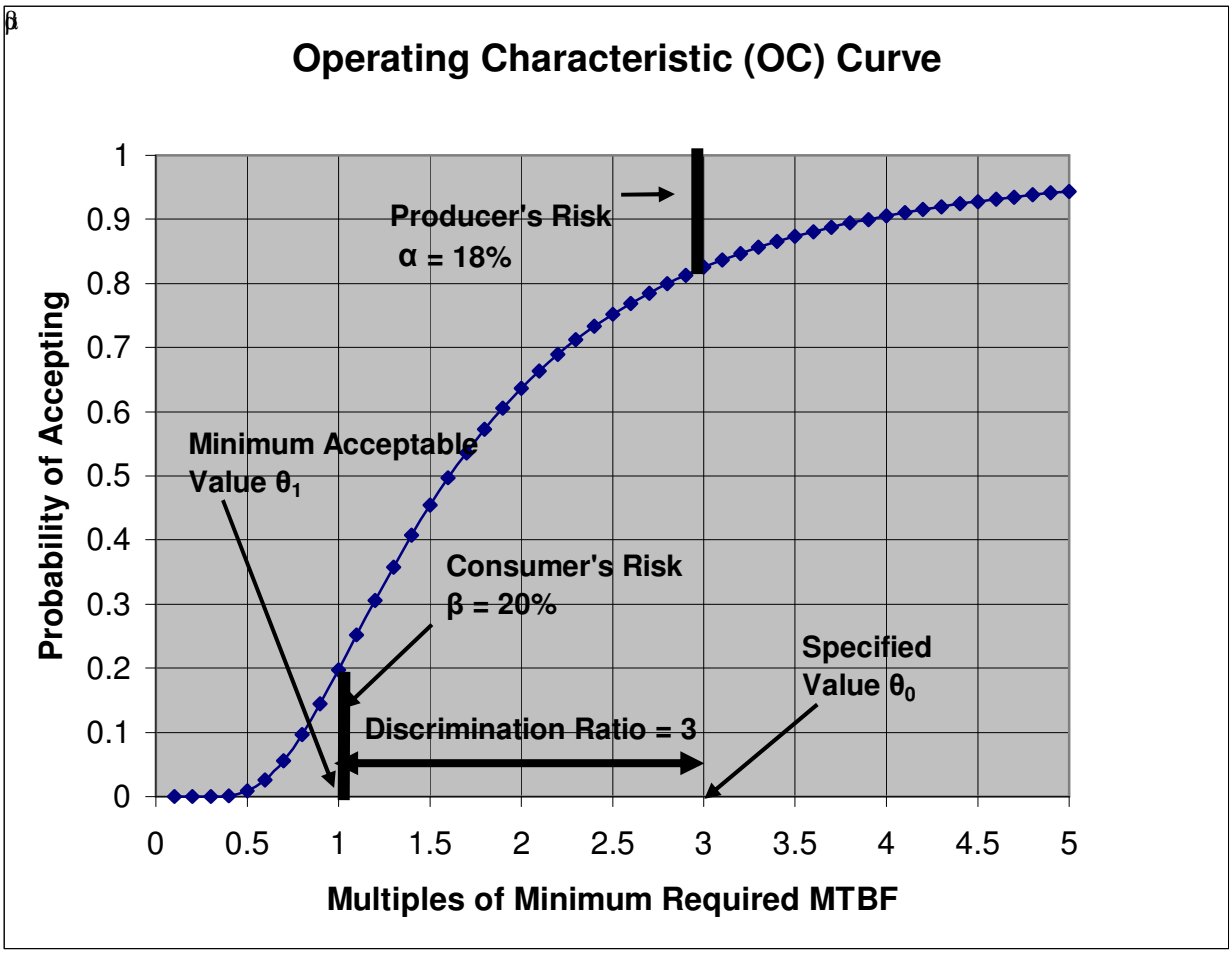

FIGURE D - 2: Risks and Decision Points Associated with OC Curve

The OC curve graphically illustrates the primary reliability test design .parameters. The region below θ_1 is the rejection region. The region above θ_0 is the acceptance region. The region in between θ_1 and θ_0 is an uncertain region in which the system is neither bad enough to demand rejection, nor good enough to demand acceptance. With a discrimination ratio of 3, the contractor still has an 18% probability of failing the reliability demonstration test even if the true MTBF of his system is three times the requirement.

In order to balance the producer's and consumer's risks, it is necessary to establish two points on the OC curve. The first point is a lower test MTBF (θ_1) that represents the minimum value acceptable to the Government. The probability of accepting a system that does not meet the FAA's minimum acceptable value is β and represents the risk (in this example 20%) to the Government of accepting a "bad" system.

The second point is the upper test MTBF (θ_0) that represents the specified value of MTBF. The probability of accepting a system at this point is (1- α) and the probability of rejecting a "good" system, α, represents the risk to the contractor, in this example, 18%.

The ratio θ_0 / θ_1 is known as the discrimination ratio, in this case, 3. This example assumes a fixed test duration of 4.3 times the lower test MTBF. To more closely approach the ideal case in the previous figure where the discrimination ratio is one and both risks are zero, the test duration must be significantly increased.

Error! Reference source not found. illustrates the effect of increasing the test duration on the OC curve. The OC curves are based on a selection of fixed length tests from MIL-STD-781. The test times associated with each of the curves expressed as multiples of the lower test MTBF are as follows:

- XVII = 4.3
- XV = 9.3
- XI = 21.1
- IX = 45

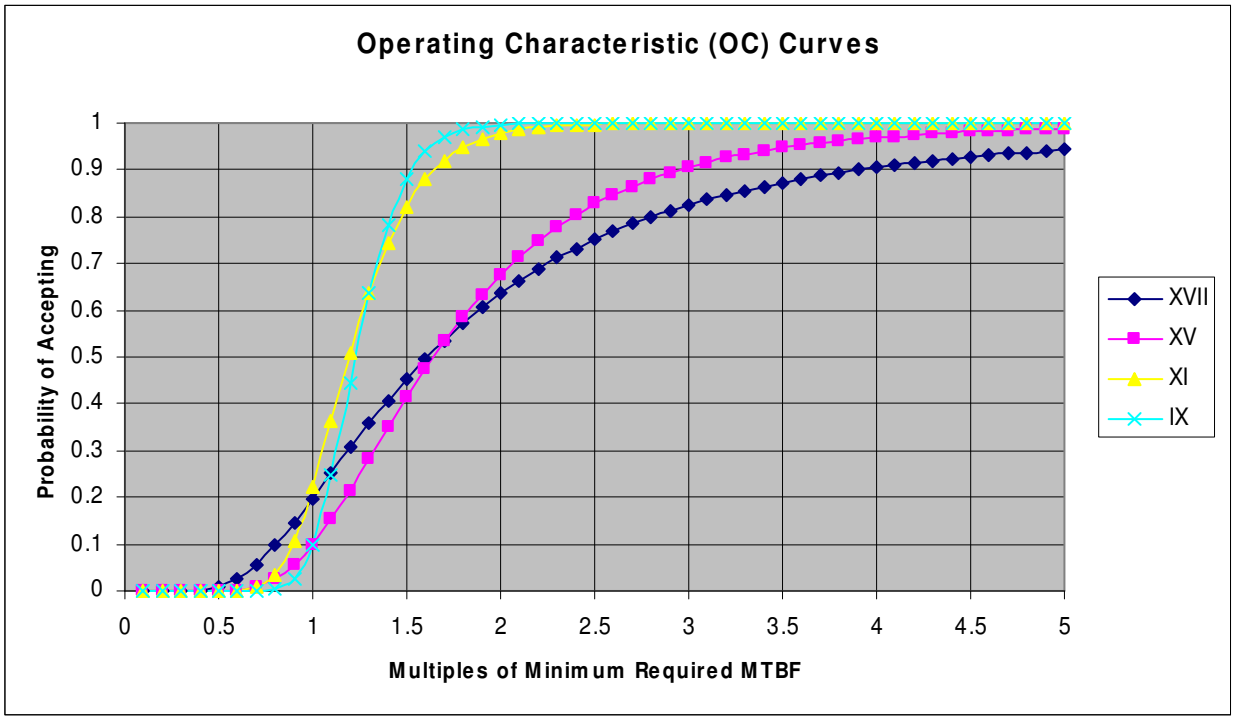

FIGURE D - 3: Effect of Increasing Test Time on OC Curve

The steepest curve has a discrimination ratio of 1.5, a consumer's risk of 9.9% and a producer's risk of 12%. These reduced risks come at a significant price however. With a test duration multiple of 45, testing a MTBF requirement of 20,000 hours for a modern fault-tolerant system would require 100 years of test exposure. This would require either testing a single system for 100 years, or testing a large number of systems for a shorter time, both of which are impractical. A general rule of thumb in reliability qualification testing is that the test time should be at least ten times the specified MTBF, which is still impractical for high reliability systems. Even the shortest test duration of the MIL-STD-781 standard test scenarios would require ten years of test time for a 20,000 hour MTBF system. Below this test duration, the test results are virtually meaningless.

While there are many different ways of looking at the statistics of this problem, they all lead to the same point: to achieve a test scenario with risk levels that are acceptable to both the Government and the contractor for a high reliability system requires an unacceptably long test time.

The above arguments are based on conventional text book statistics theory. They do not address another practical reality: Modern software systems are not amenable to fixed reliability qualification tests. Software is dynamic; enhancements, program trouble reports, patches, and so on presents a dynamic, ever changing situation that must be effectively managed. The only practical alternative is to pursue an aggressive reliability growth program and deploy the system to the field only when it is more stable than the system it will replace. Formal reliability demonstration programs such as those used for the electronic "black boxes" of the past are no longer feasible.

The OC curves in the charts in this appendix are based on standard fixed length test scenarios from MIL-STD-781 and calculated using Excel functions as described below.

The statistical properties of a fixed duration reliability test are based on the Poisson distribution. The lower tail cumulative Poisson distribution is given by

$$P(ac \mid \theta) = \sum_{k=0}^{c} \frac{\left(\frac{T}{\theta}\right)^{k} e^{-\left(\frac{T}{\theta}\right)}}{k!} \qquad [D\text{-}1]$$

Where

P(ac|θ) = the probability of accepting a system whose true MTBF is θ.

c = maximum acceptable number of failures

θ = True MTBF

θ_0 = Upper test MTBF

θ_1 = Lower test MTBF

T = Total test time

The Excel statistical function "POISSON (x, mean, cumulative)" returns the function in Equation [C-1] when the following parameters are substituted in the Excel function:

POISSON (c, T/θ, TRUE)

The charts were calculated using the fixed values for c and T associated with each of the sample test plans from MIL-STD-781. The x axis of the charts is normalized to multiples of the lower test MTBF, θ_1, and covers a range of 0.1 to 5.0 times the minimum MTBF acceptable to the Government.

Appendix E SERVICE THREAD DIAGRAMS

The figures in this Appendix represent the Service Thread diagrams for the Service Threads defined in TABLE 6-2. Each of the figures contains a box showing the NAS architecture capabilities supported by the Service Thread. The numbers in parentheses after each capability represent the STLSC value associated with the Service Thread's support of that capability as shown in the cells associated with that Service Thread column in the STLSC matrices in **FIGURE 6-7**, FIGURE 6-8, and FIGURE 6-9. Capabilities with STLSC value of "N" for the Service Thread are not listed on the diagrams. (A cell entry of "N" for "not rated" indicates one of two conditions: (1) Loss of the capability is overshadowed by the loss of a much more critical capability, which renders the provision of the capability meaningless in that instance, or (2) The capability is used very infrequently, and should not be treated as a driver for RMA requirements.)

Some of the Service Thread diagrams are simply a single box with an arrow output. In some cases, this is because the service is provided by a single component or module so that the single component is in fact the entire Service Thread. In other cases, either the specific architecture of the Service Thread has not been defined or there is insufficient information about it to complete the Service Thread diagram. In these cases, the Service Thread diagram serves primarily as a placeholder that can be elaborated upon as more information about the architecture of the Service Thread becomes available.

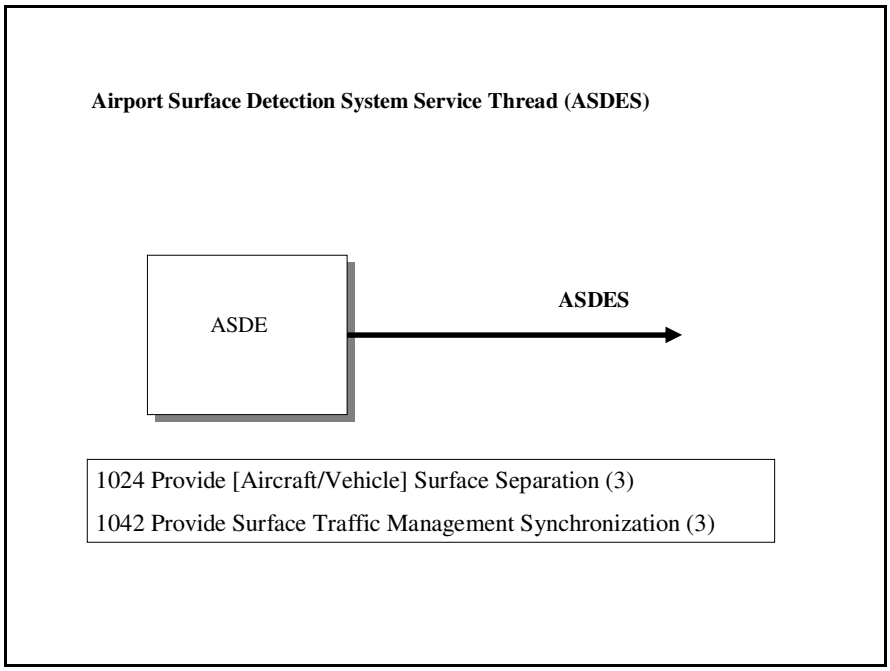

FIGURE E - 1: Airport Surface Detection Equipment (ASDES)

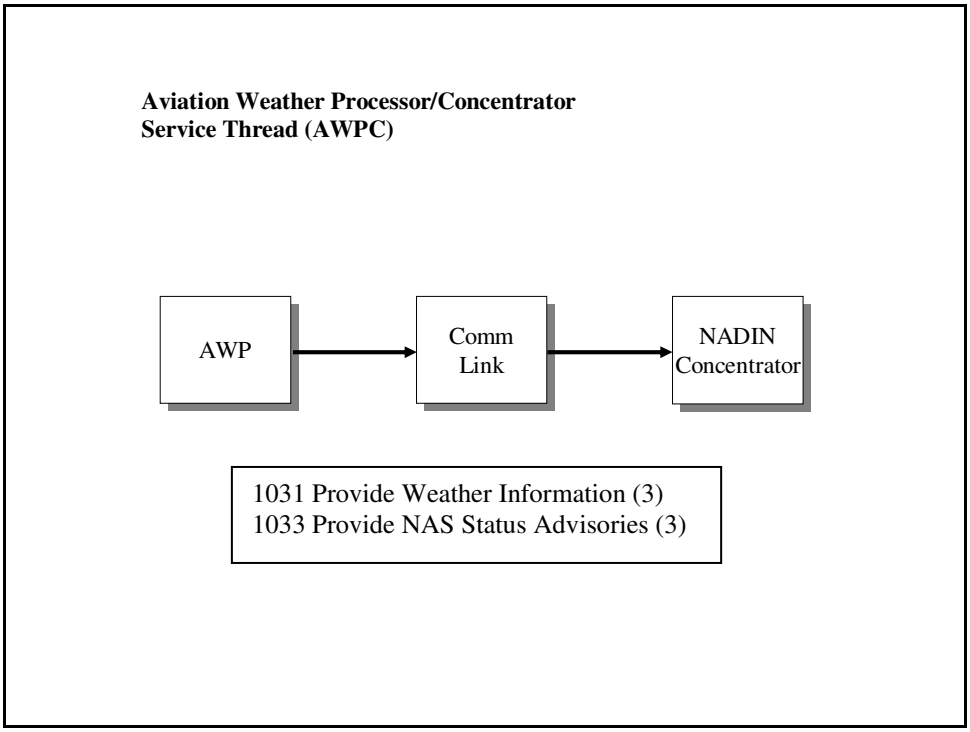

FIGURE E - 2: Aviation Weather Processor Concentrator (AWPC

)

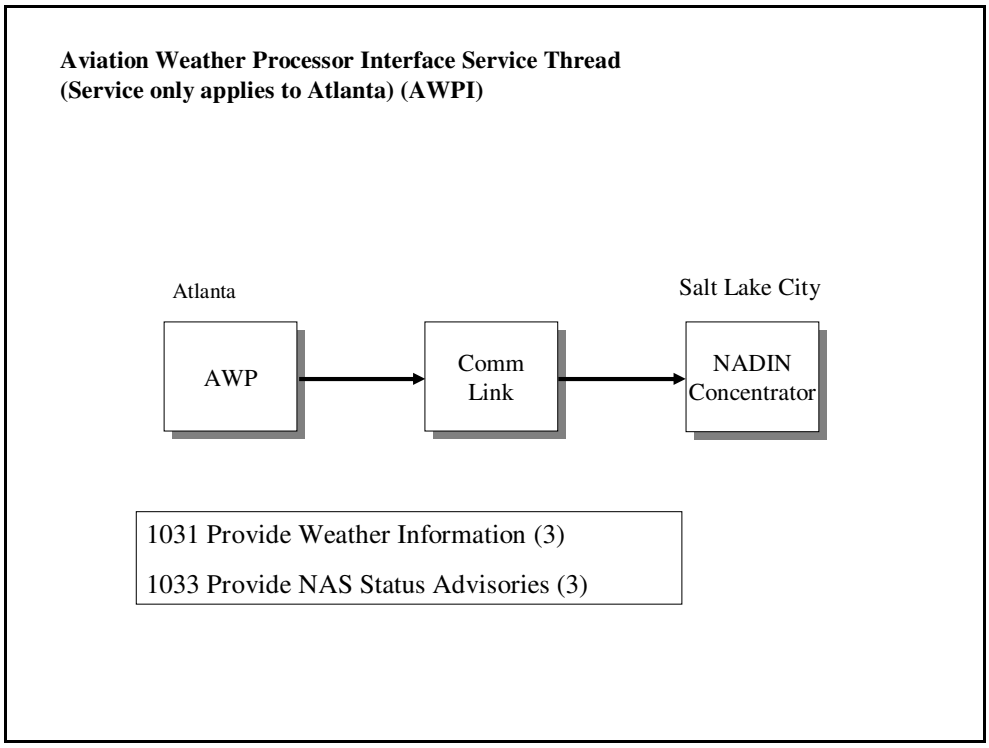

FIGURE E - 3 : Aviation Weather Processor Interface (AWPI)

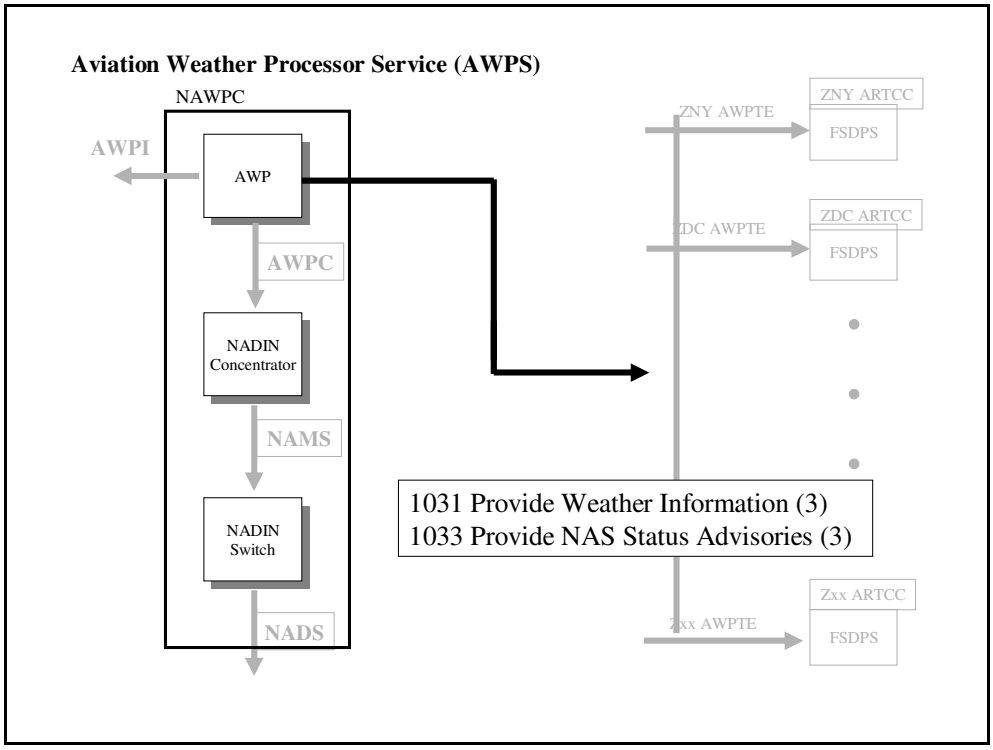

FIGURE E - 4: Aviation Weather Processor Service (ASPS)

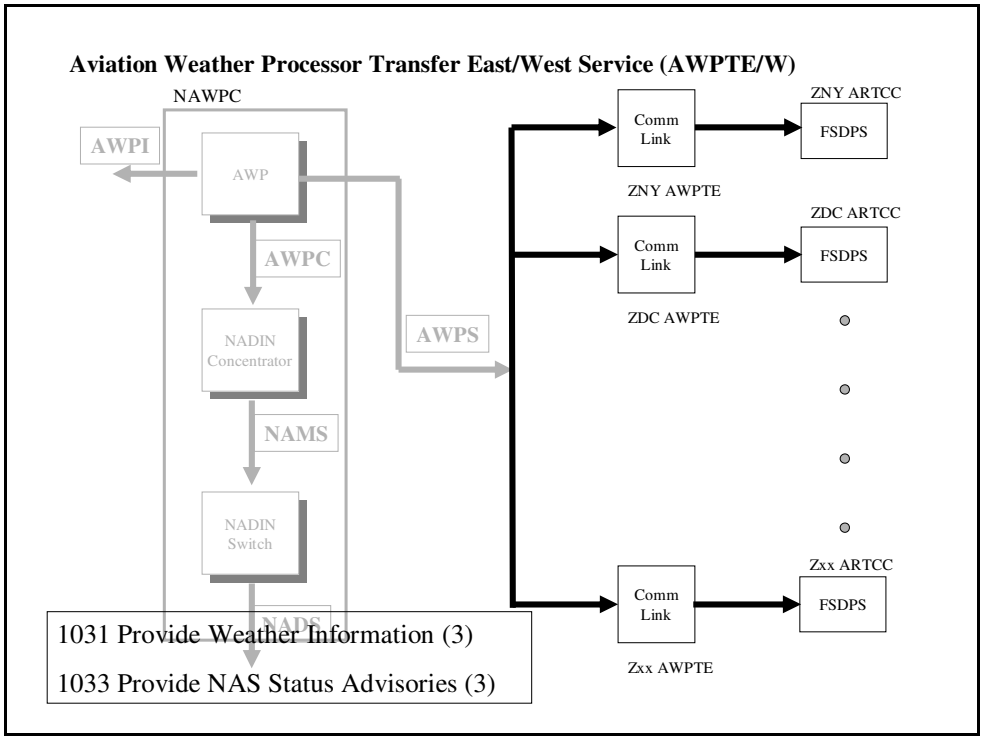

FIGURE E - 5 : Aviation Weather Processor Transfer East/West Service (AWPTE/W)

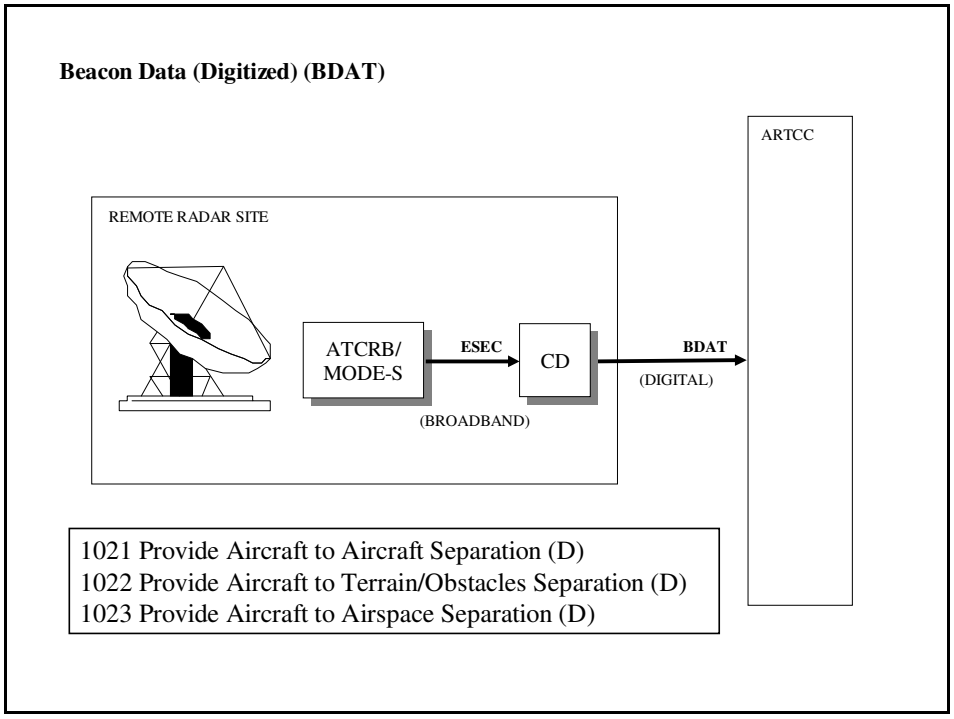

FIGURE E - 6 : Beacon Data (Digitized) (BDAT)

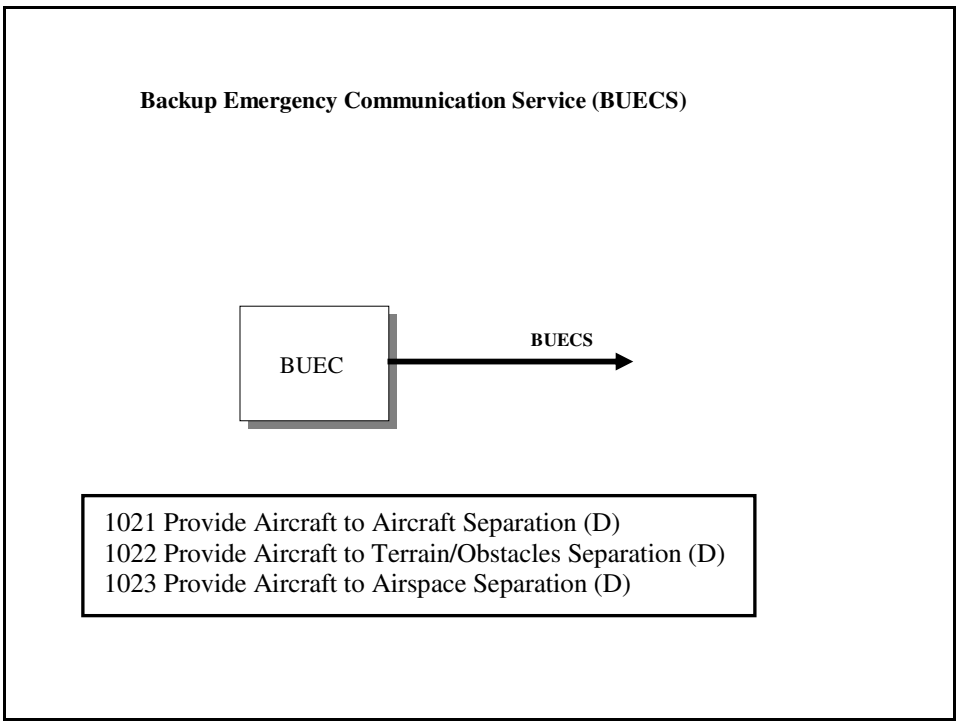

FIGURE E - 7: Backup Emergency Communications Service (BUECS)

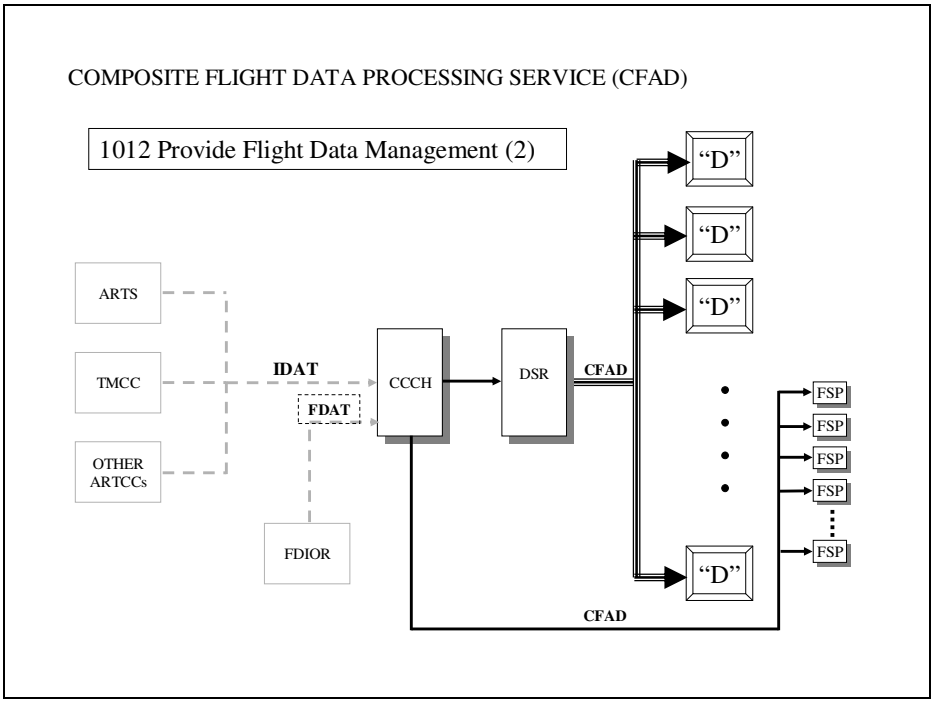

FIGURE E - 8 : Composite Flight Data Processing (CFAD)

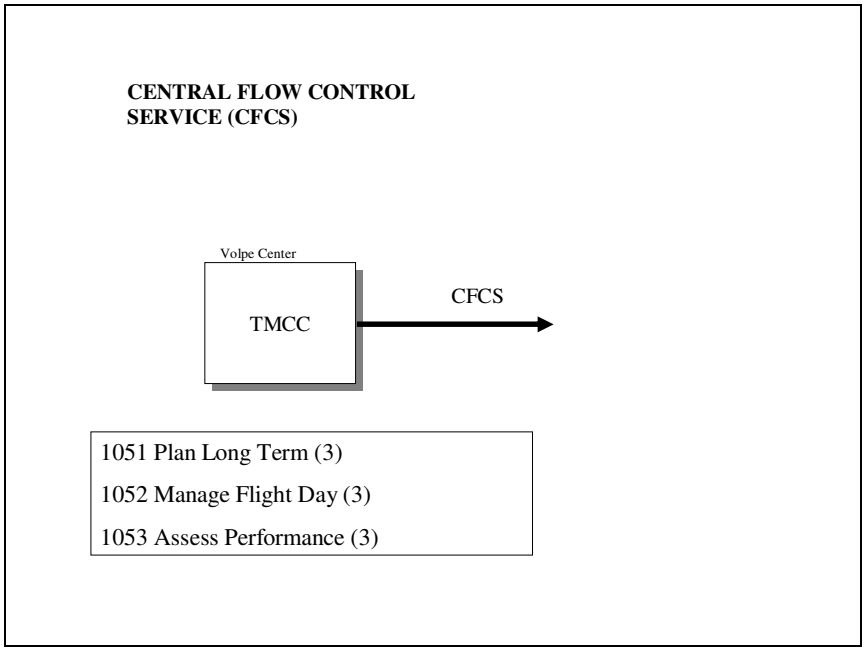

FIGURE E - 9: Central Flow Control Service (CFCS)

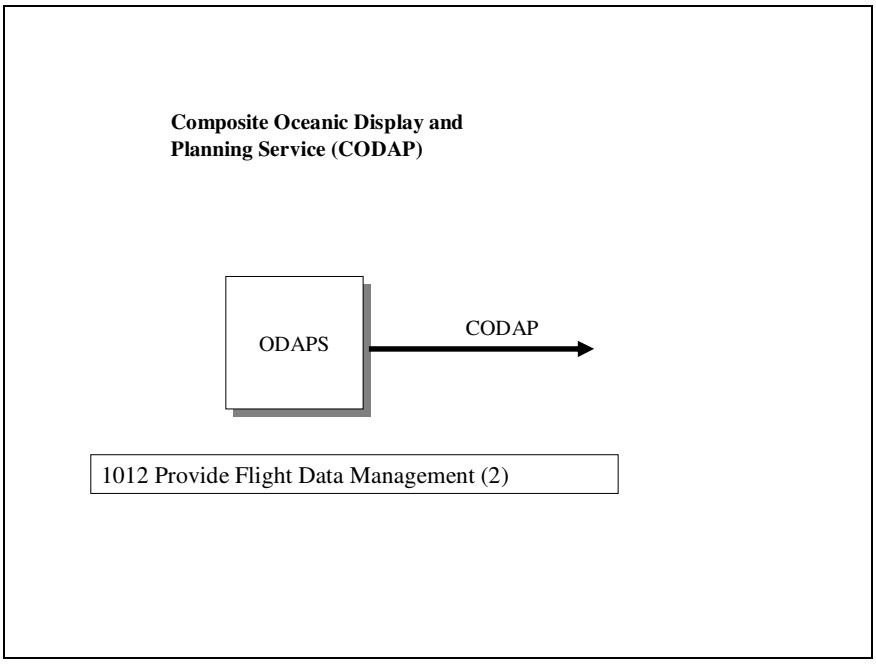

FIGURE E - 10 : Composite Oceanic Display and Planning Service (CODAP)

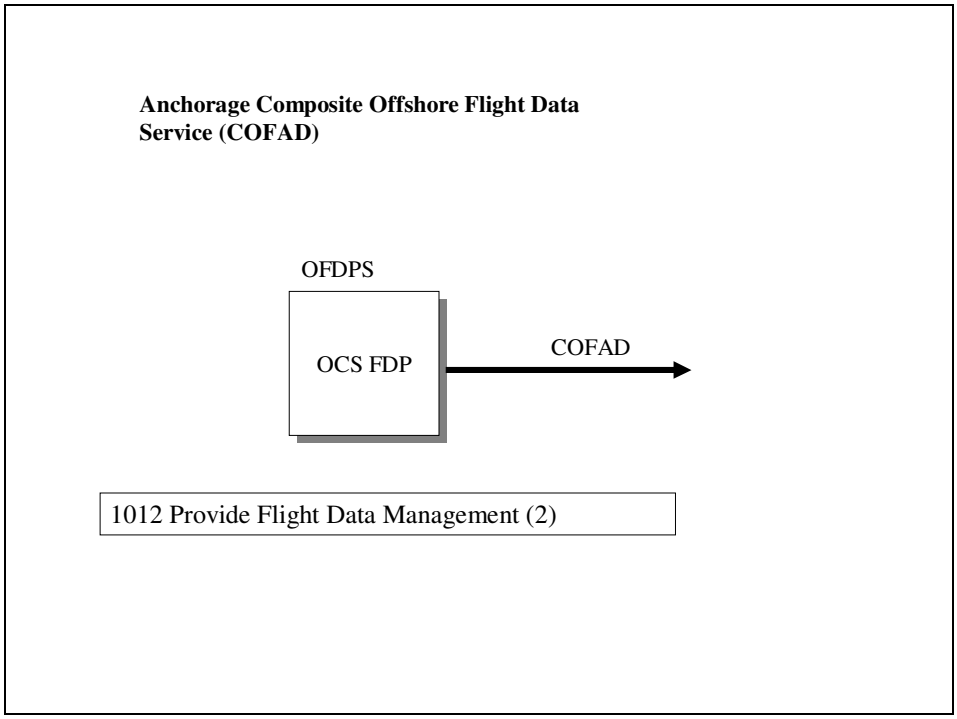

FIGURE E - 11 : Anchorage Composite Offshore Flight Data Service (COFAD)

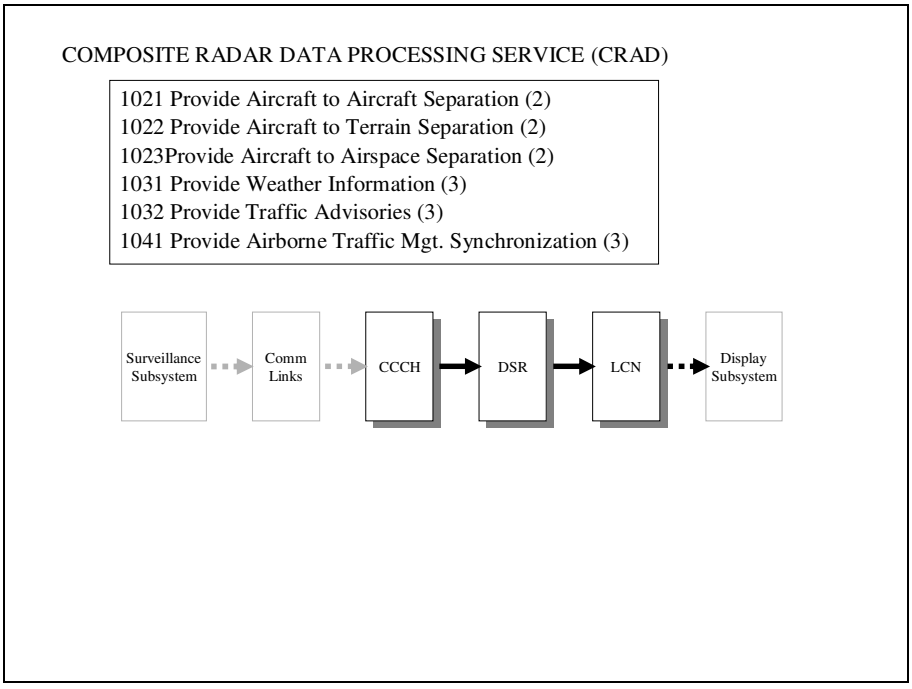

FIGURE E - 12: Composite Radar Data Processing

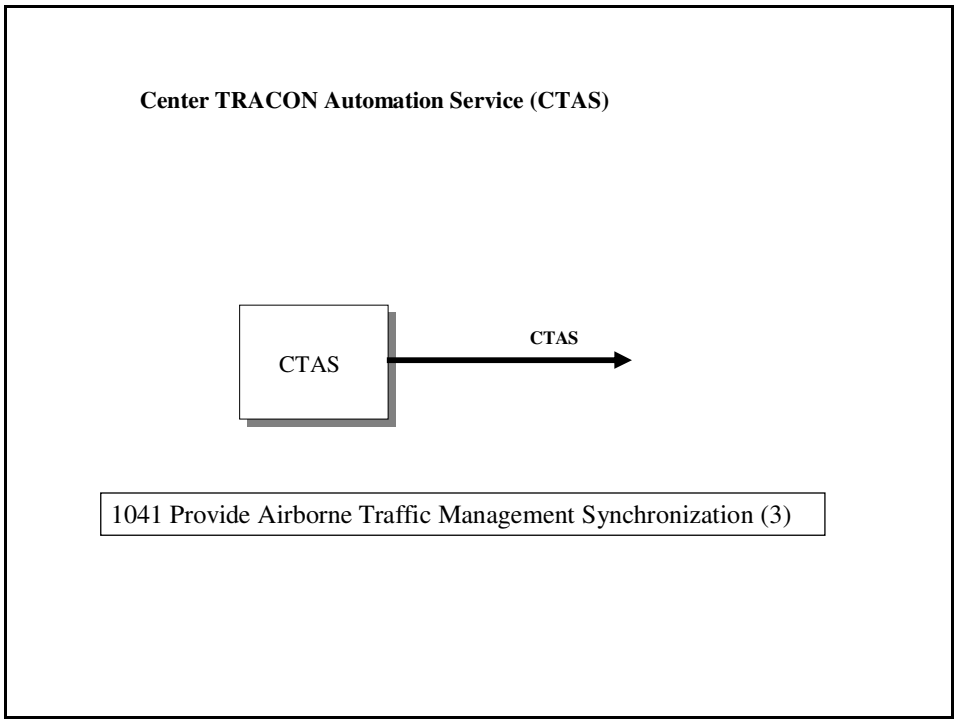

FIGURE E - 13: Center TRACON Automation System

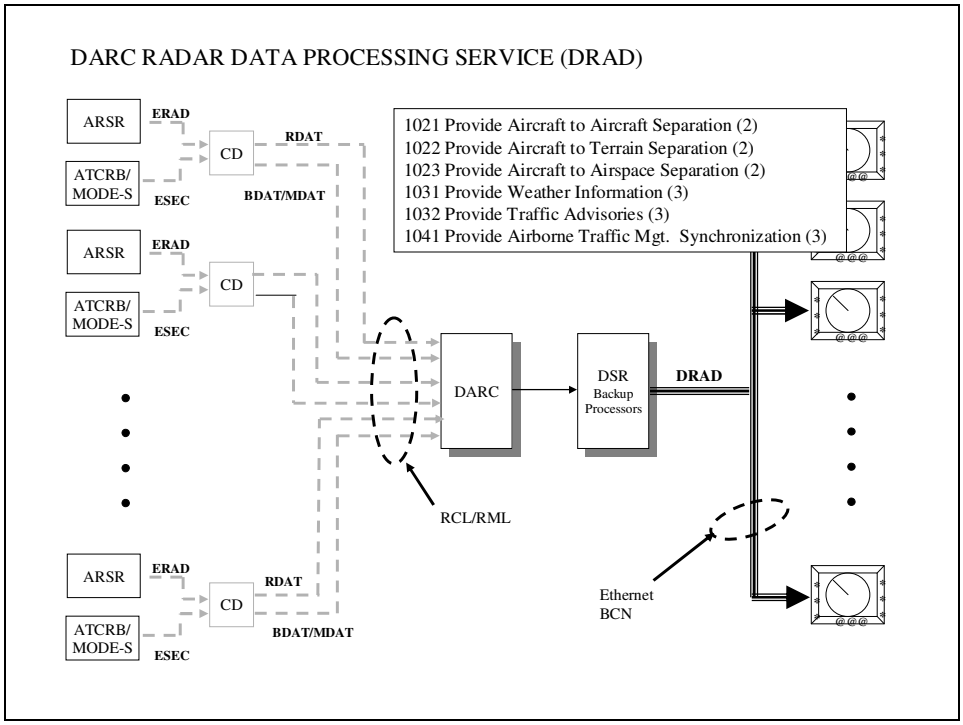

FIGURE E - 14: DARC Radar Data Processing Service (DRAD)

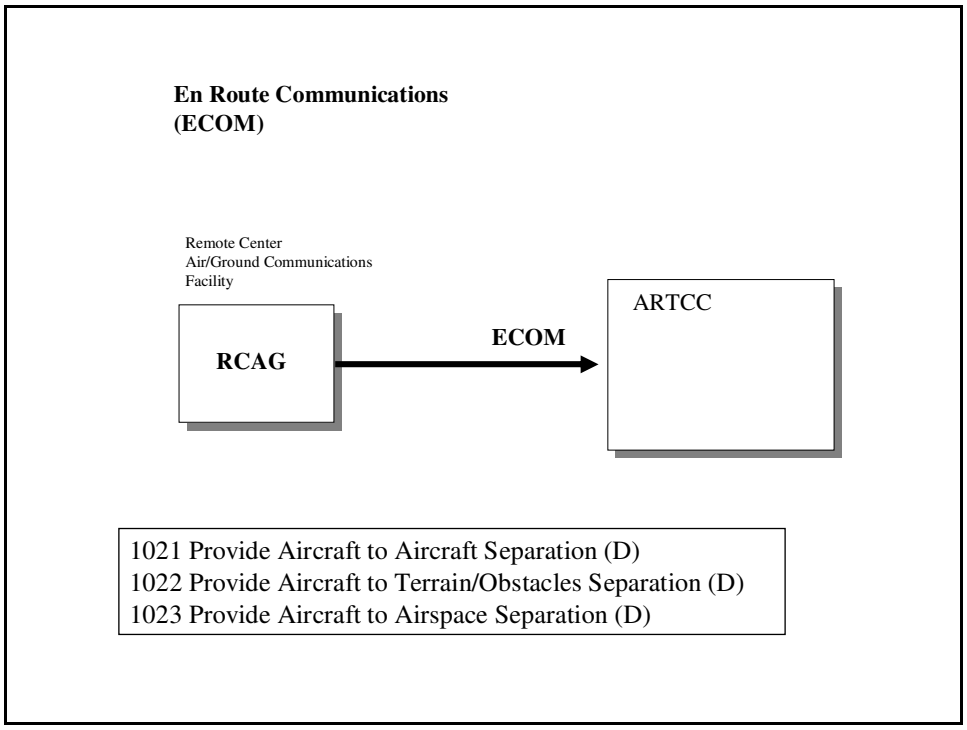

FIGURE E - 15: En Route Communications (ECOM)

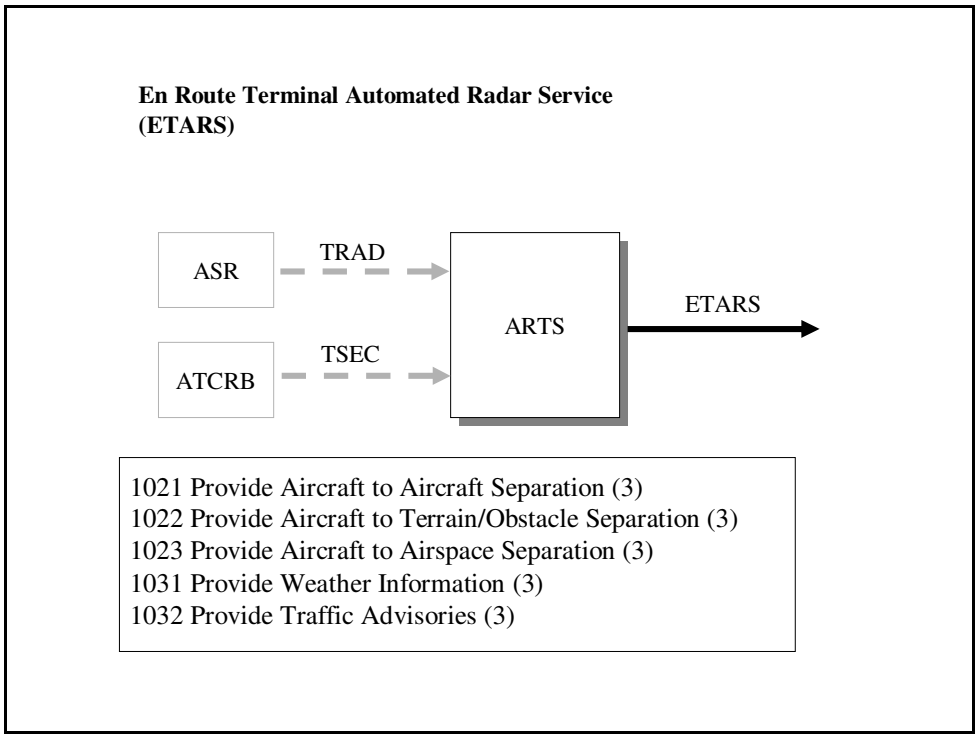

FIGURE E - 16 : En Route Terminal Automated Radar Service (ETARS)

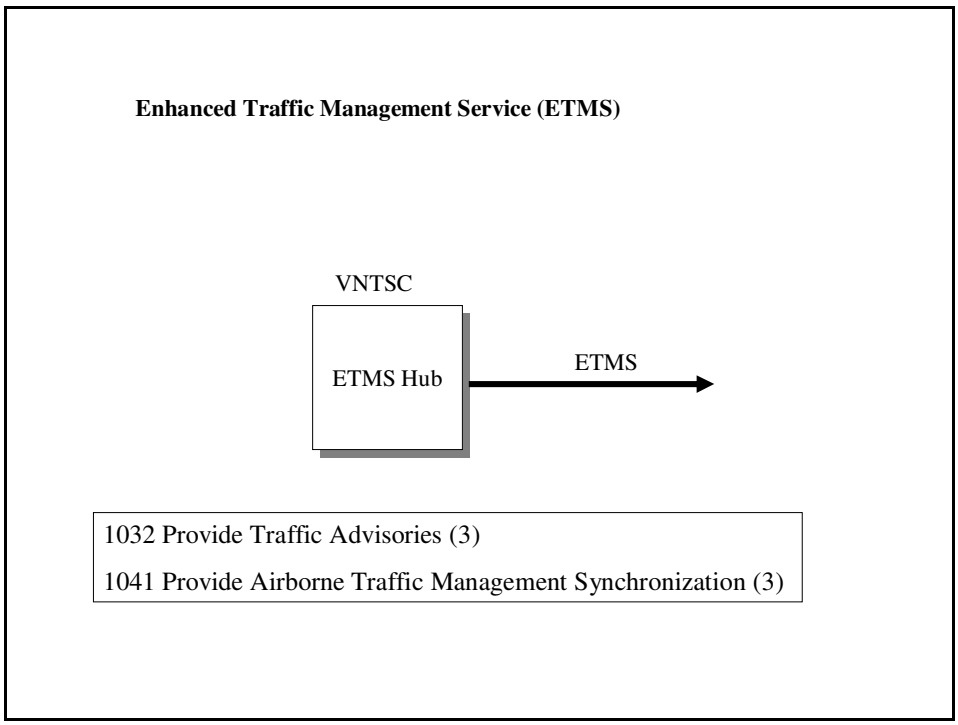

FIGURE E - 17 : Enhanced Traffic Management System (ETMS)

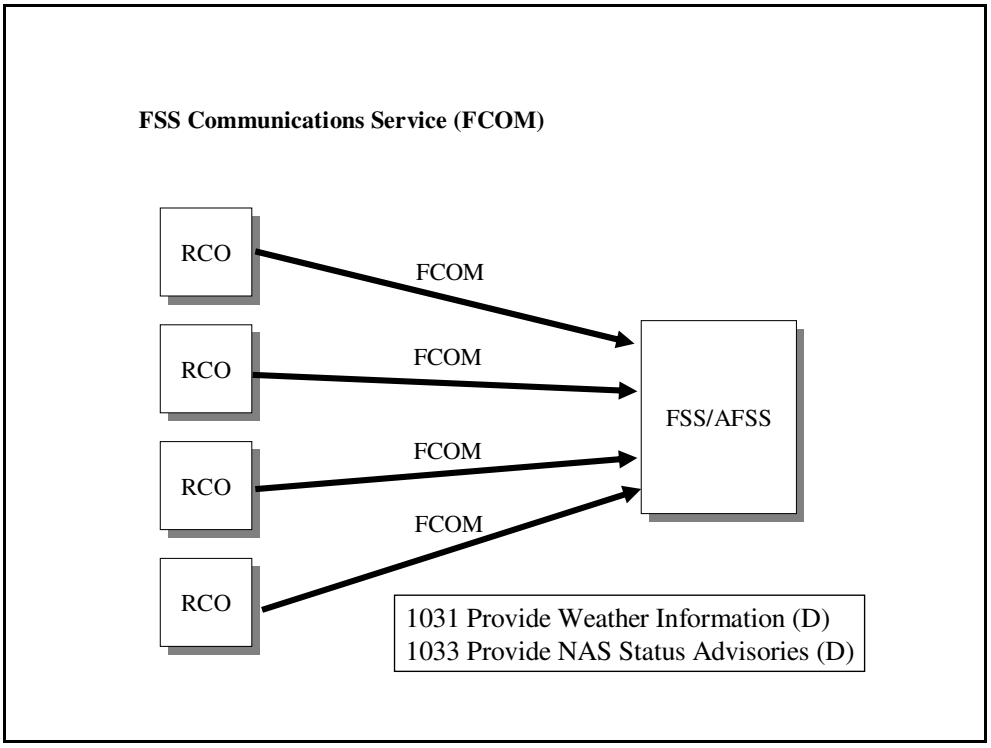

FIGURE E - 18: FSS Communications Service (FCOM)

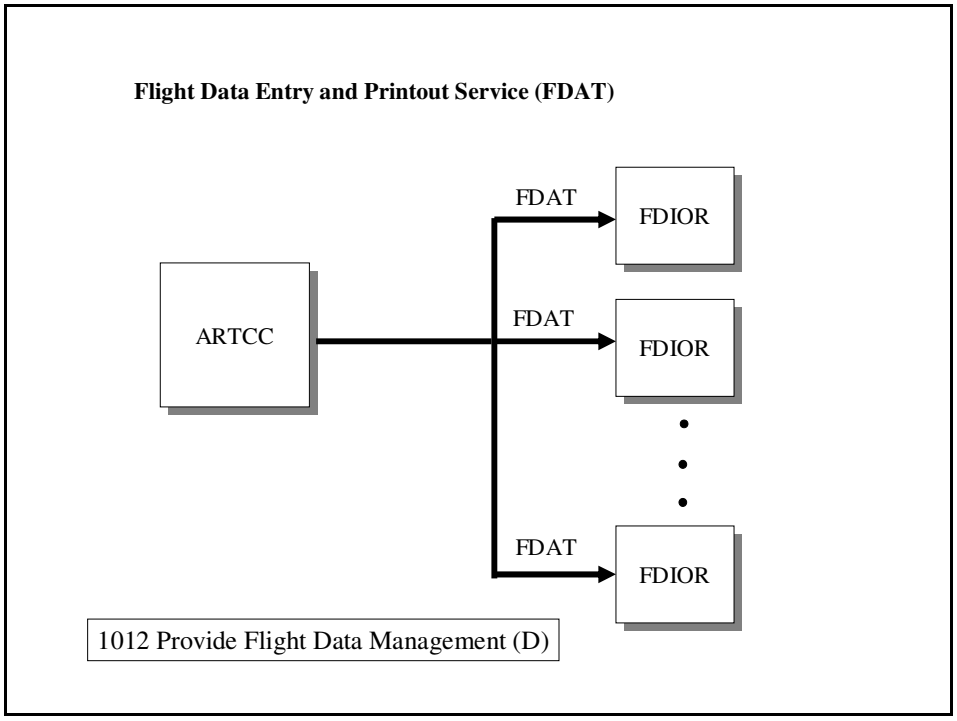

FIGURE E - 19: Flight Data Entry and Printout Service

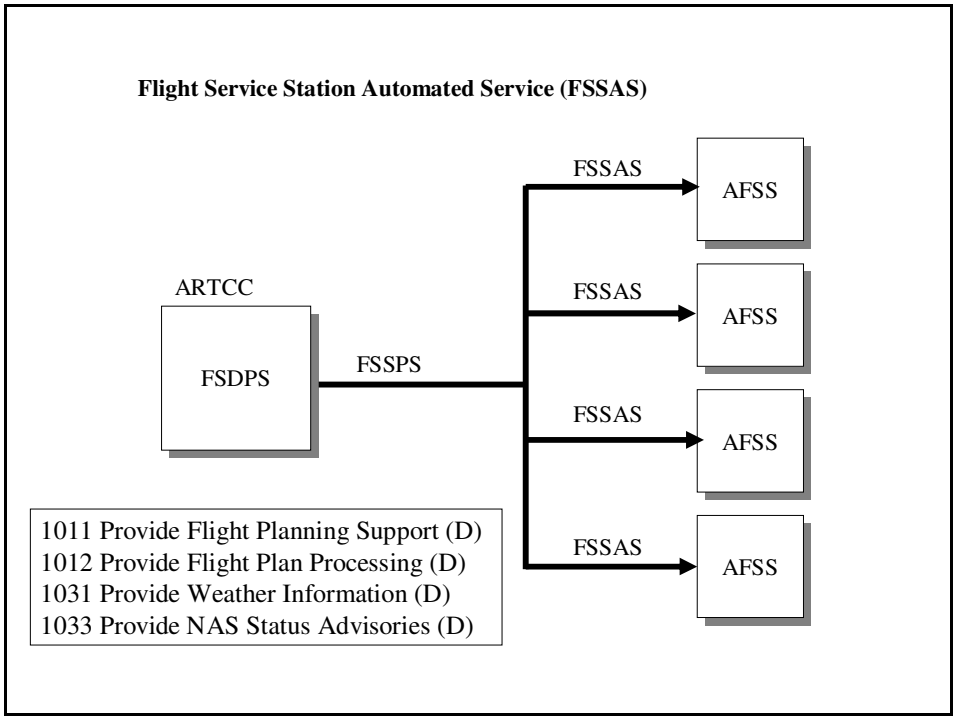

FIGURE E - 20 Flight Service Station Automated Service (FSSAS)

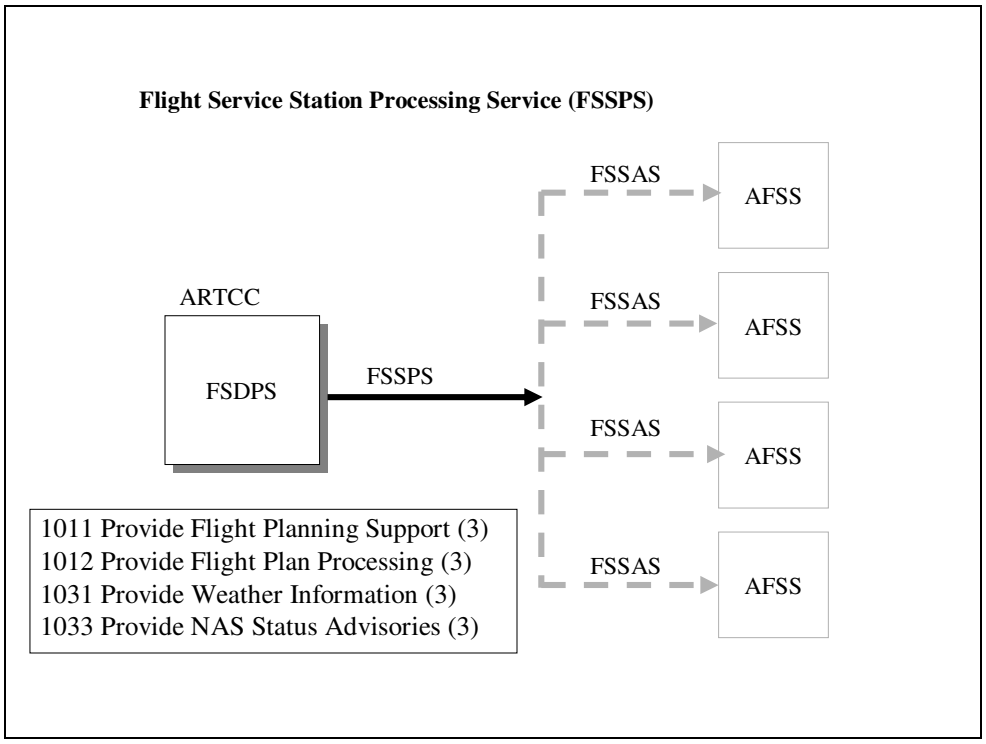

FIGURE E - 21 : Flight Service Station Processing Service (FSSPS)

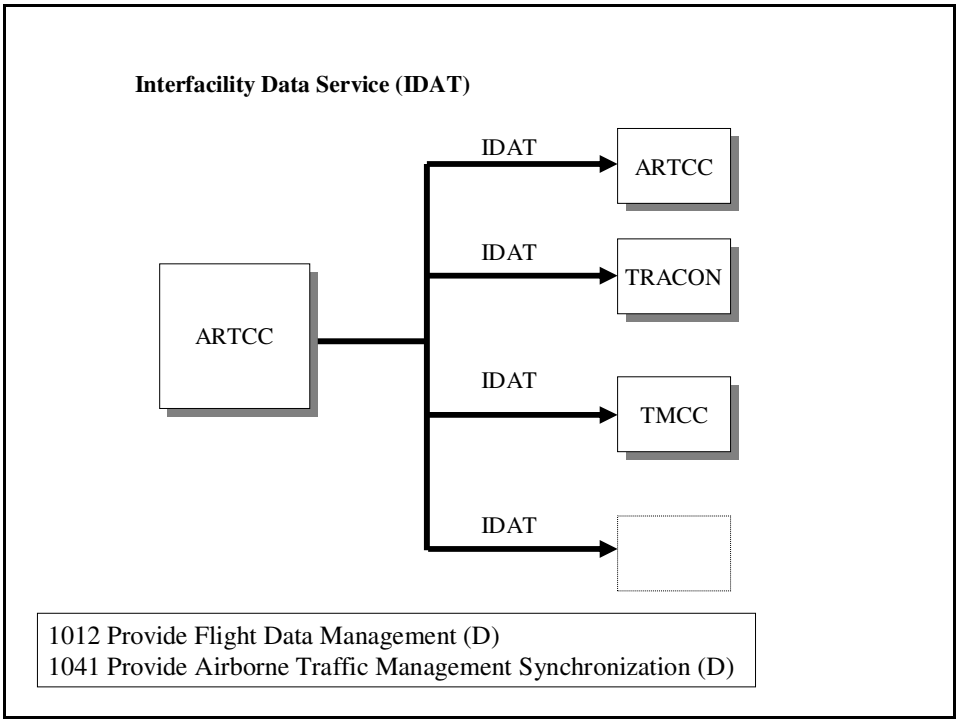

FIGURE E - 22: Interfacility Data Service (IDAT)

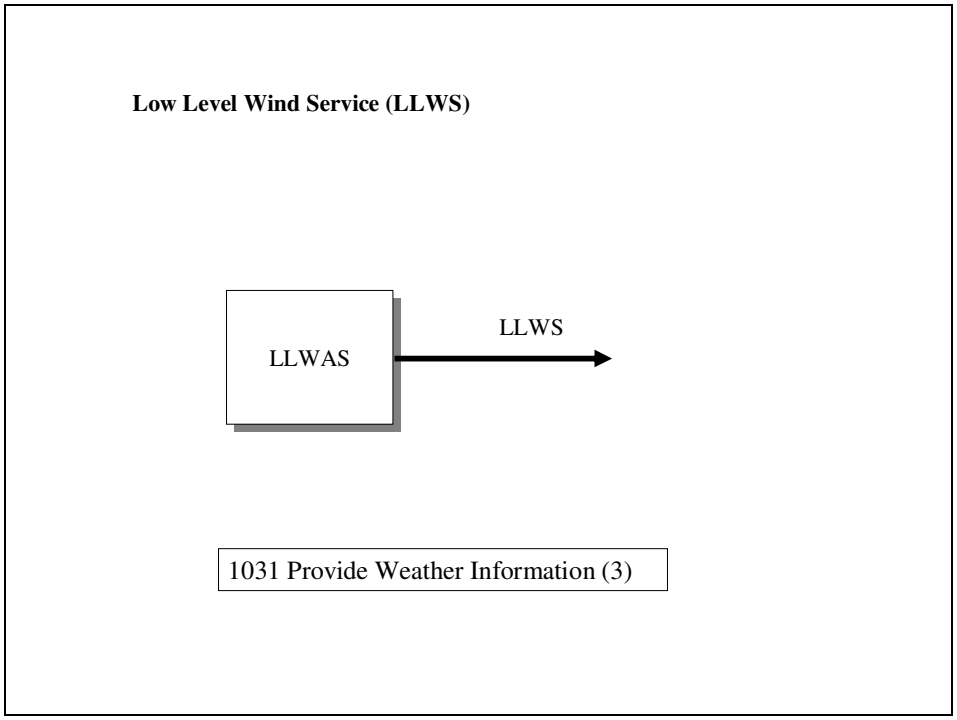

FIGURE E - 23: Low Level Wind Service (LLWS)

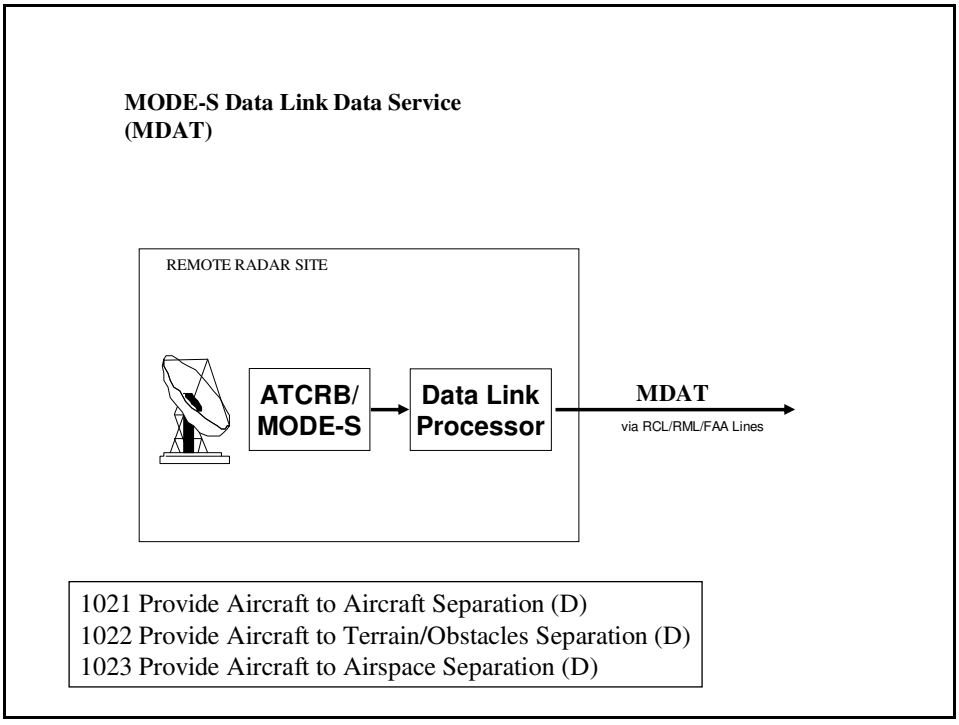

FIGURE E - 24: MODE S Data Link Data Service (MDAT)

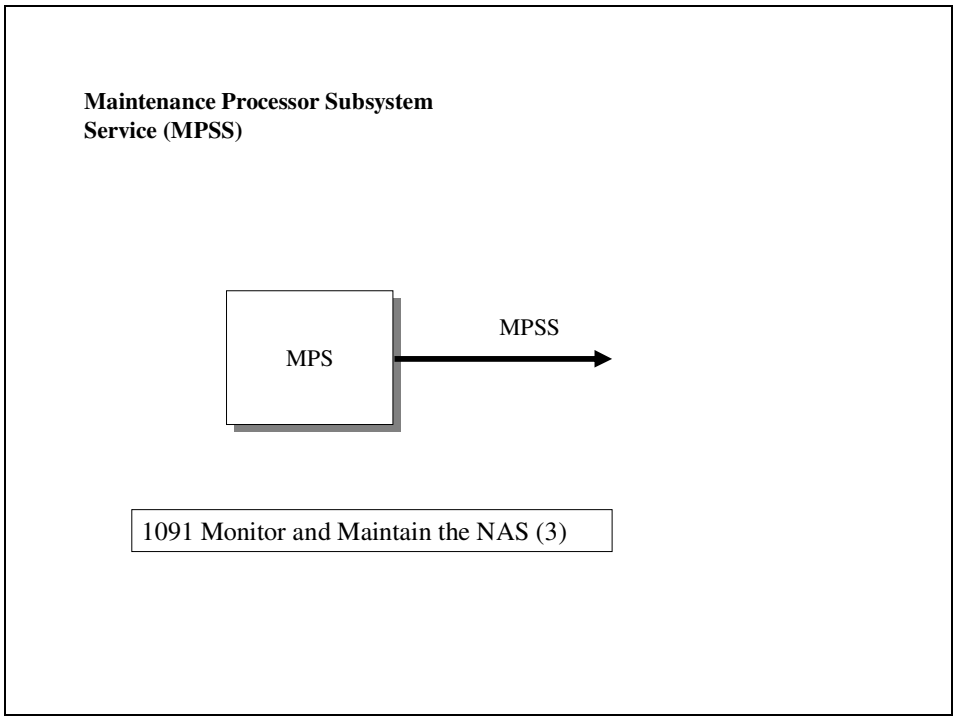

FIGURE E - 25 : Maintenance Processor Subsystem (MPSS)

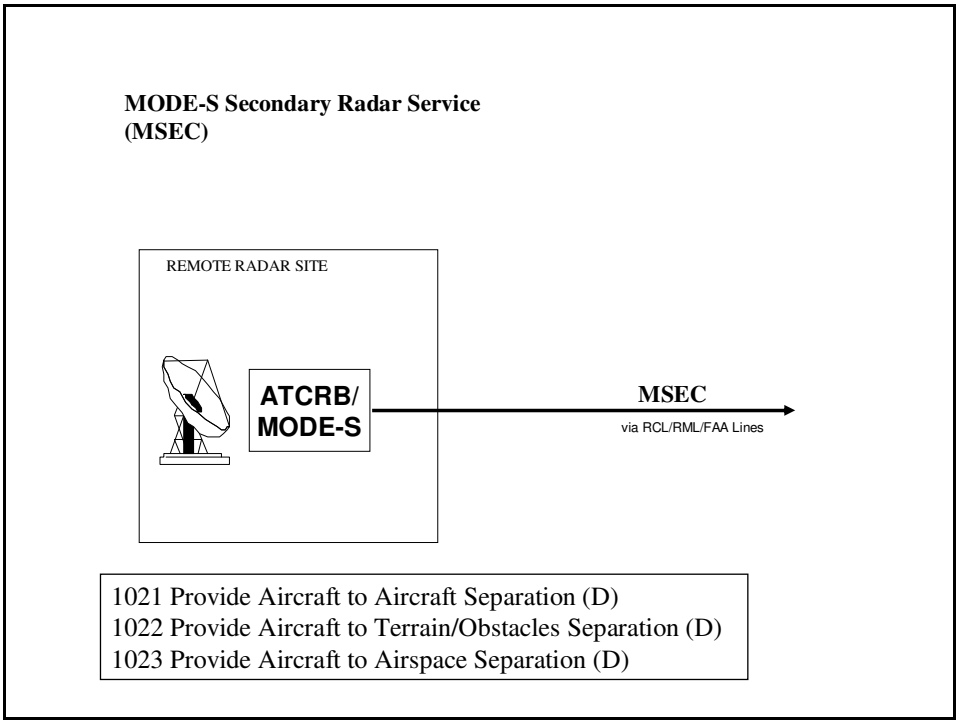

FIGURE E - 26: MODE S Secondary Radar Service (MSEC)

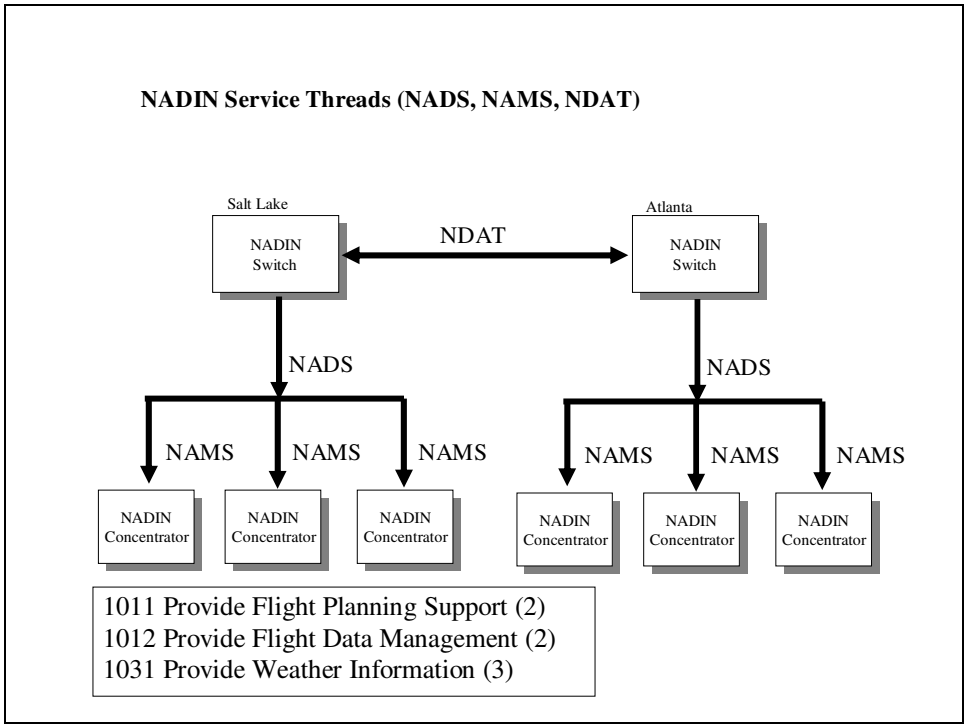

FIGURE E - 27: NADIN Service Threads

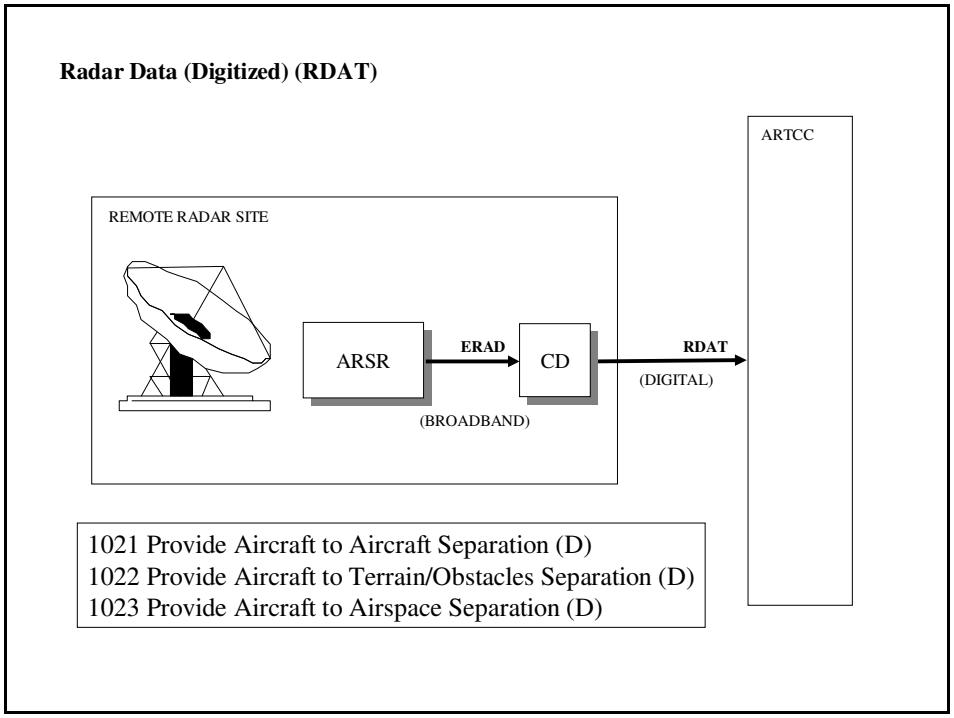

FIGURE E - 28: Radar Data (Digitized) (RDAT)

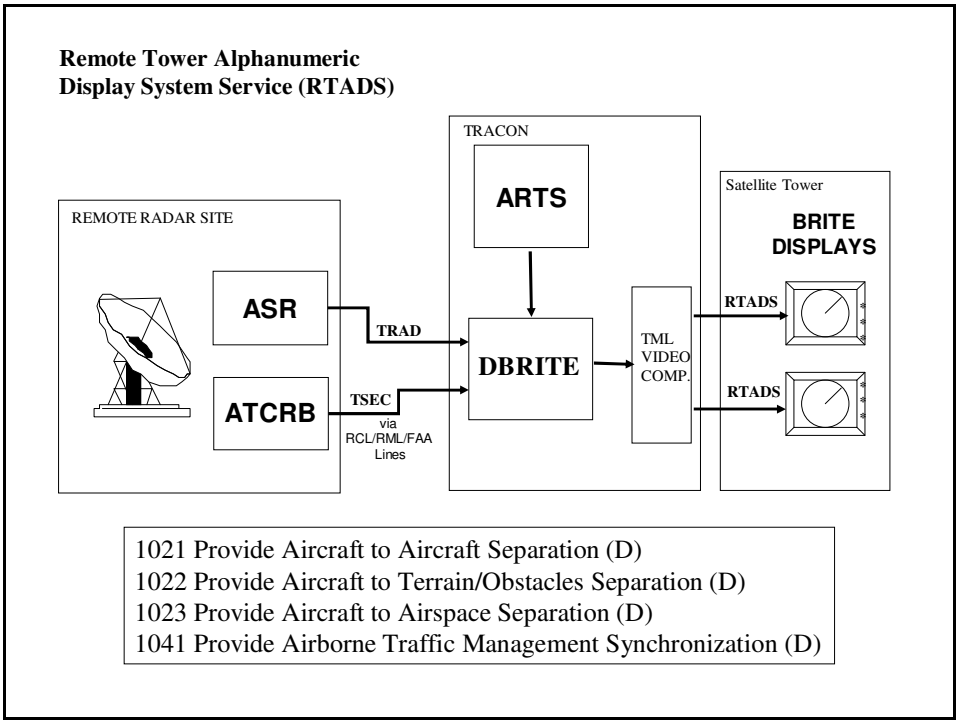

FIGURE E - 29: Remote Tower Alphanumeric Display System Service (RTADS)

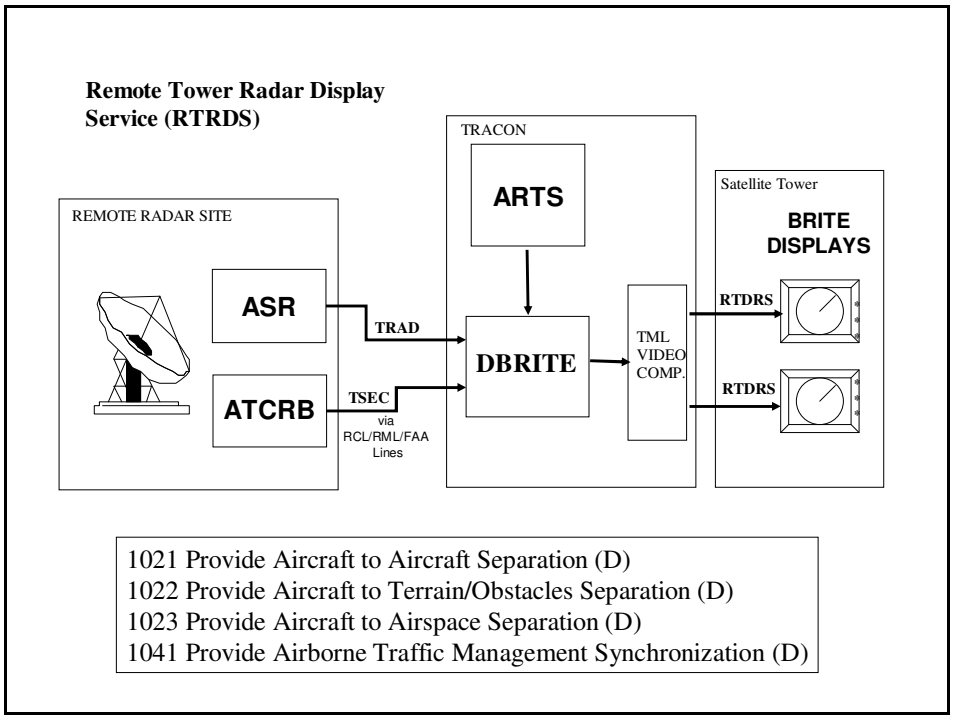

FIGURE E - 30: Remote Tower Radar Display Service (RTRDS)

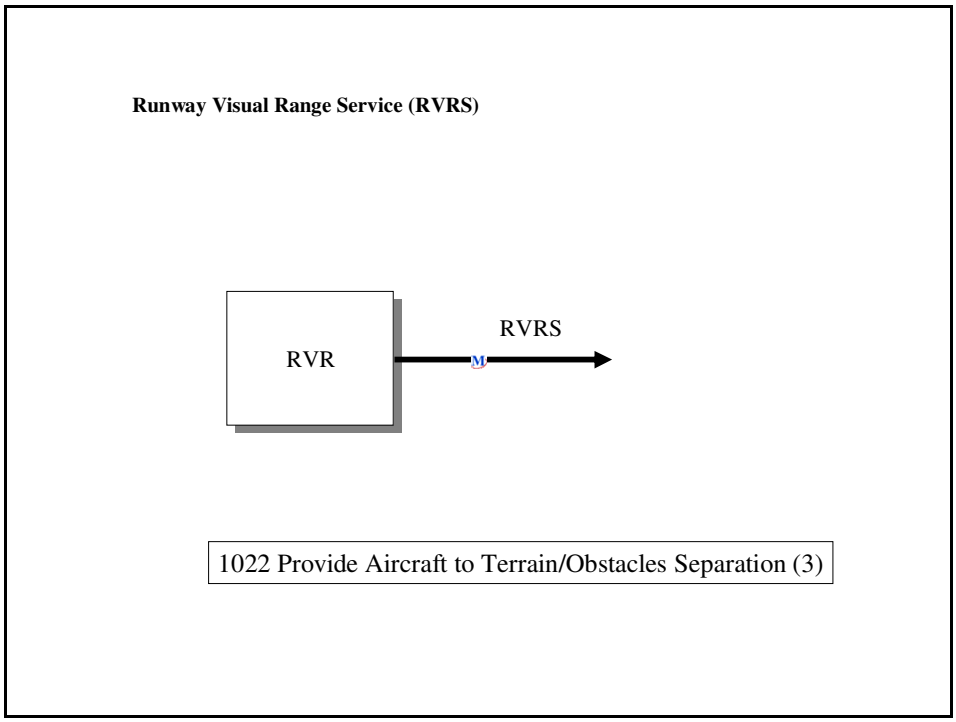

FIGURE E - 31: Runway Visual Range Service (RVRS)

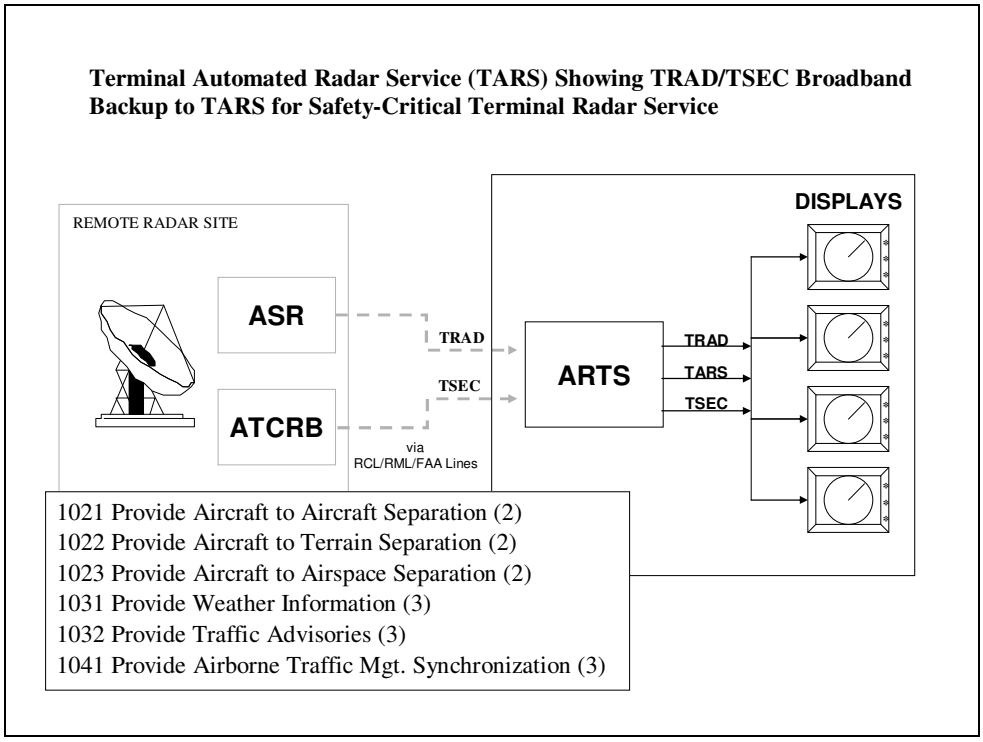

FIGURE E - 32: Terminal Automated Radar Service (TARS)

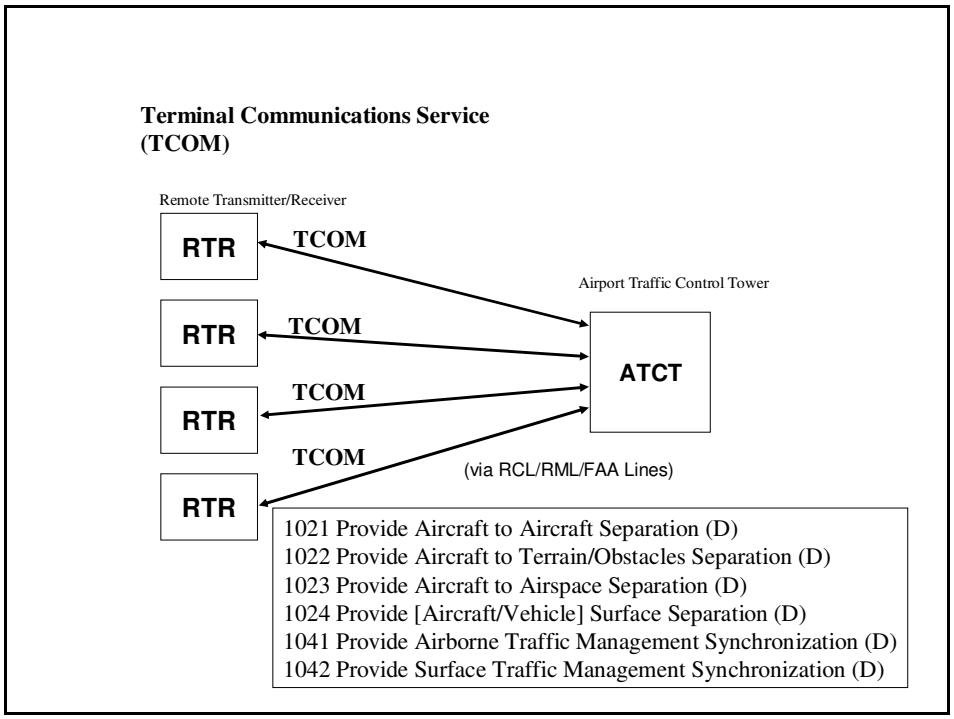

FIGURE E - 33: Terminal Communications (TCOM)

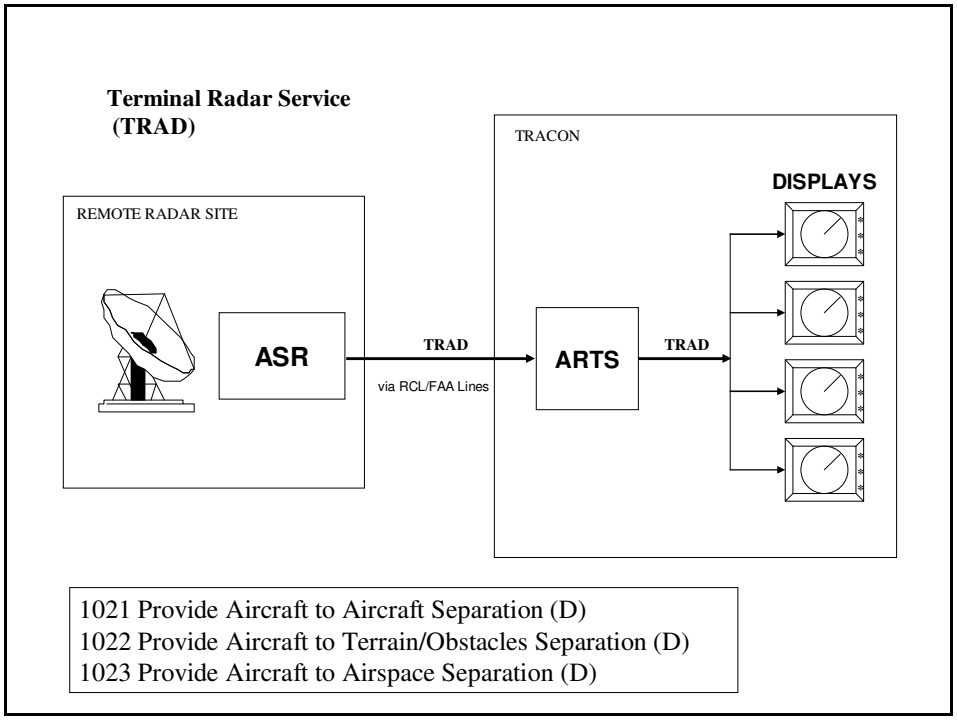

FIGURE E - 34: Terminal Radar Service (TRAD)

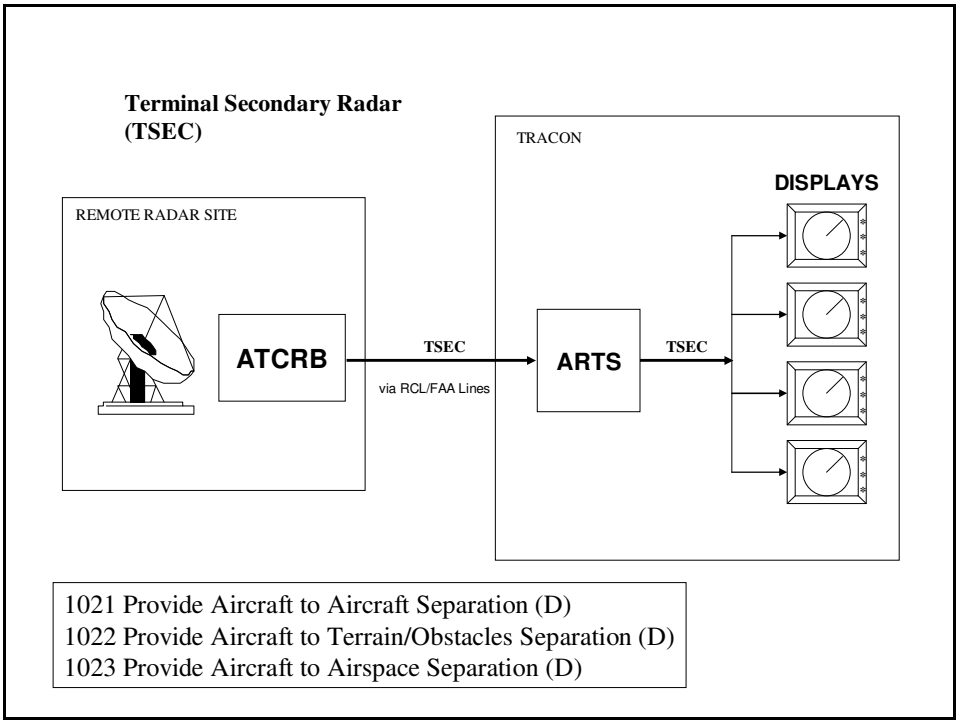

FIGURE E - 35: Terminal Secondary Radar (TSEC)

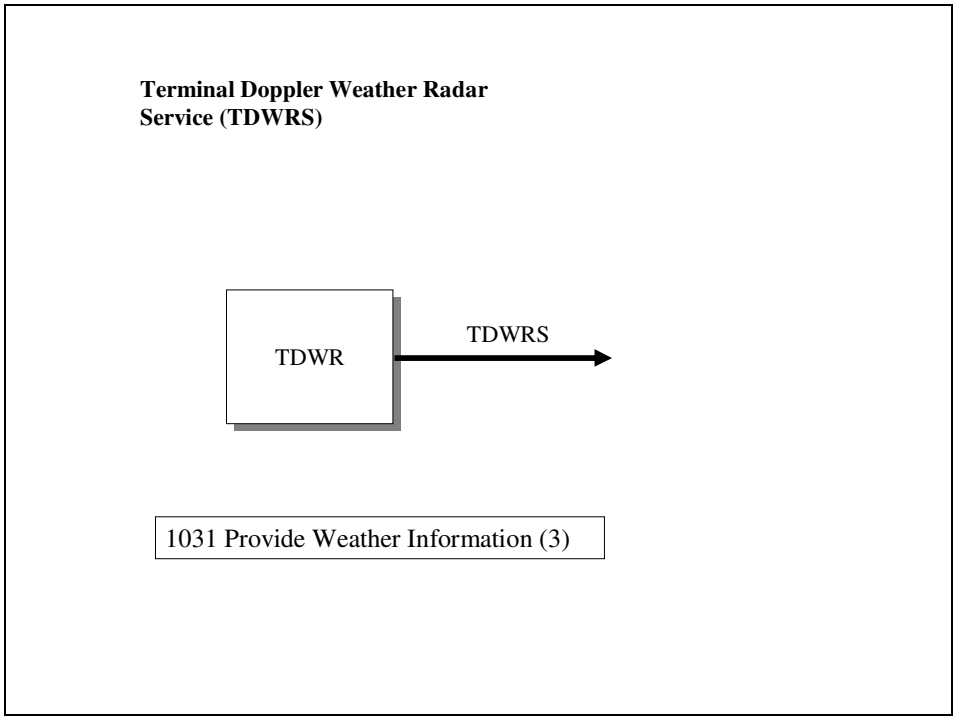

FIGURE E - 36: Terminal Doppler Weather Radar Service (TDWRS)

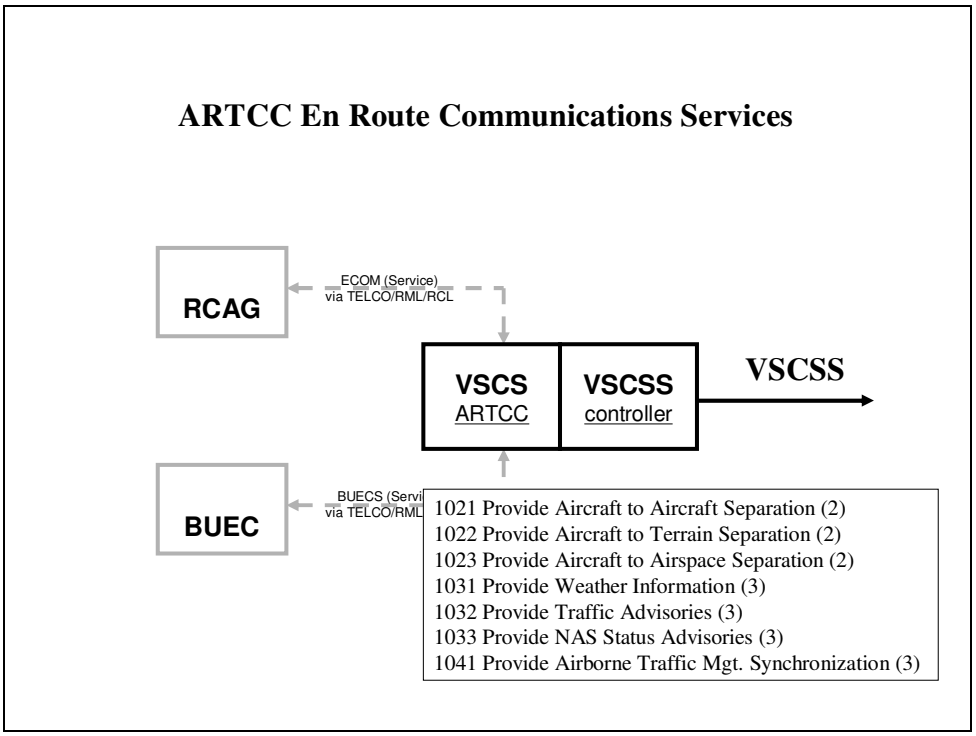

FIGURE E - 37: Voice Switching and Control System Service (VSCSS)

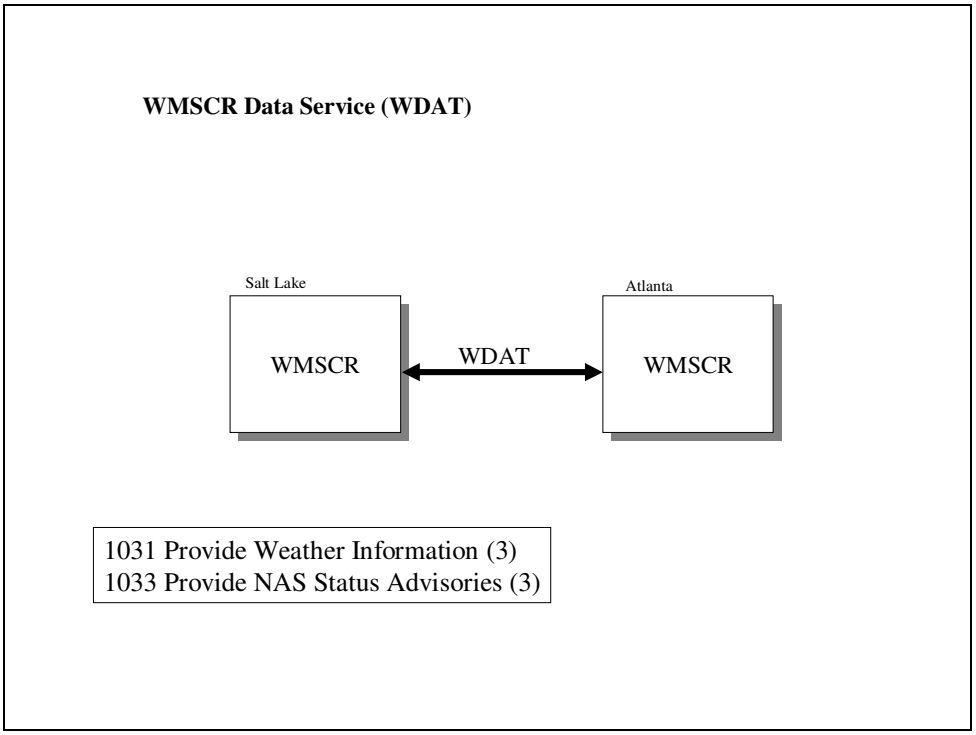

FIGURE E - 38 WMSCR Data Service (WDAT)

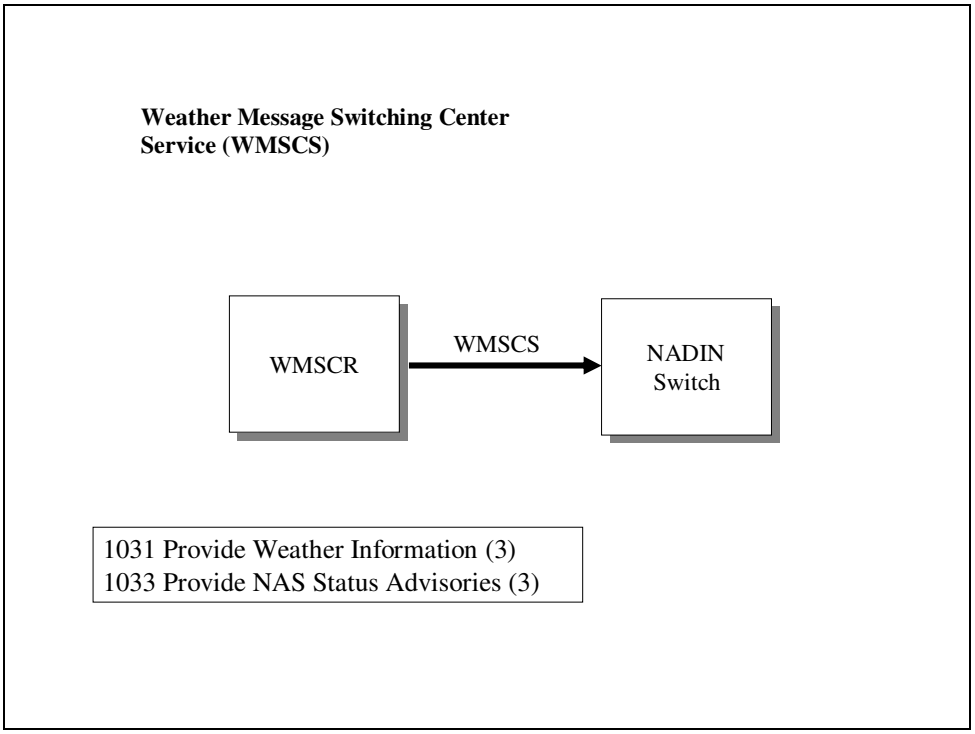

FIGURE E - 39: Weather Message Switching Center (WMSCS)

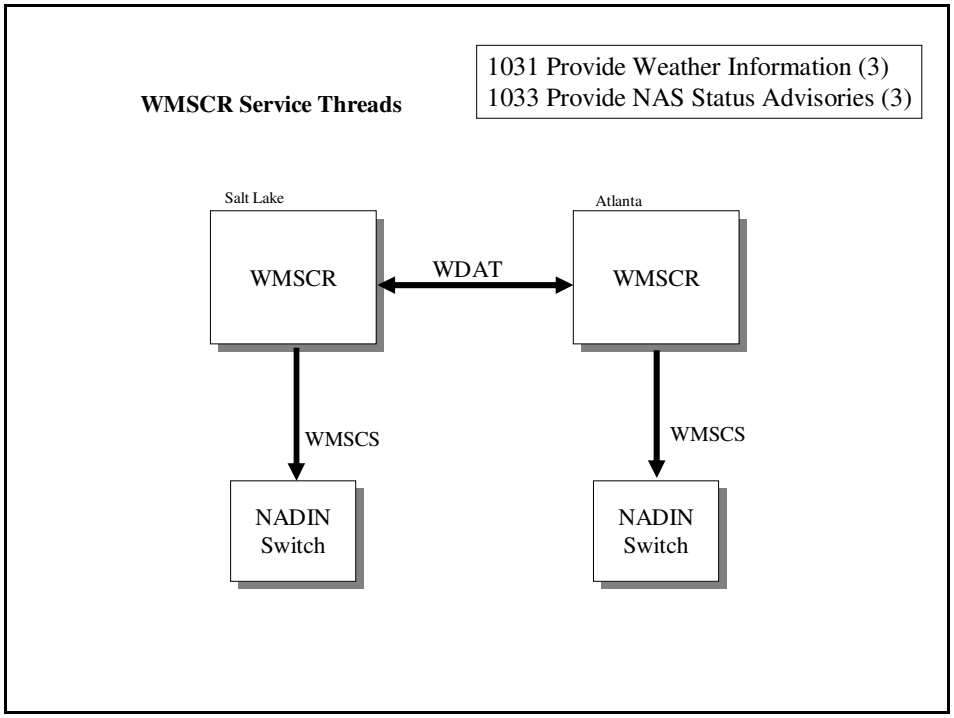

FIGURE E - 40: Weather Message Switching Center Replacement (WMSCR) Service Threads

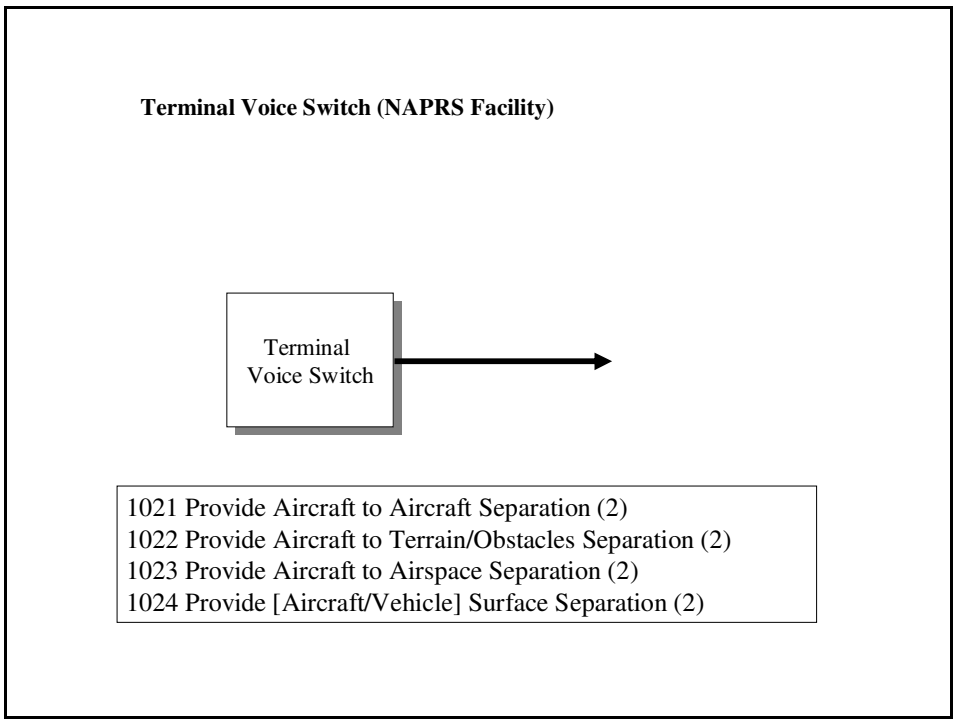

FIGURE E - 41: Terminal Voice Switch Service Thread

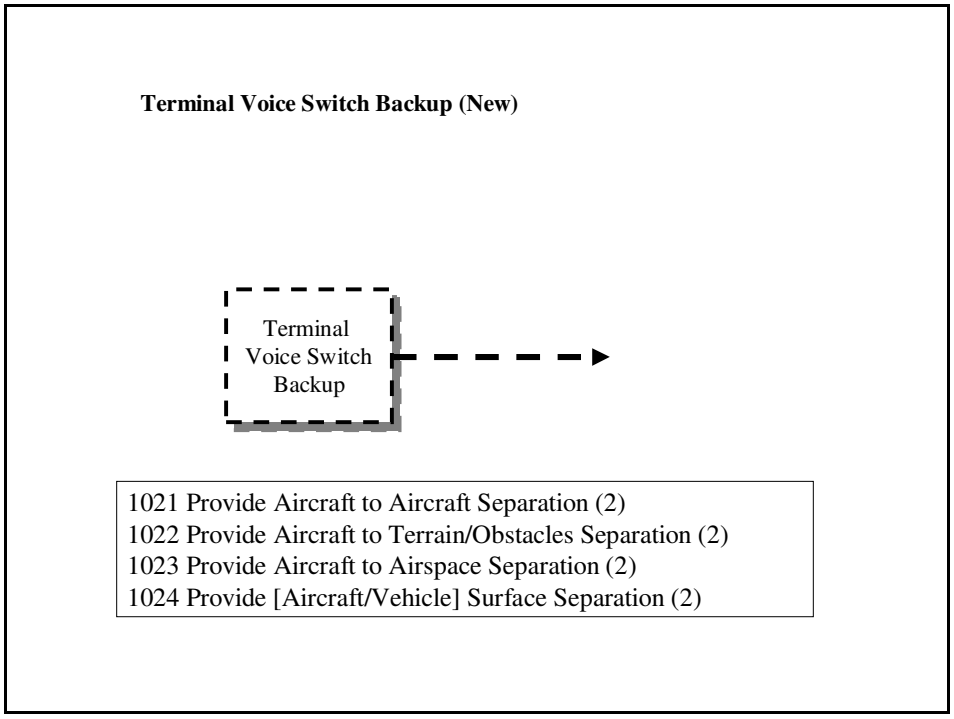

FIGURE E - 42: Terminal Voice Switch Backup (New)

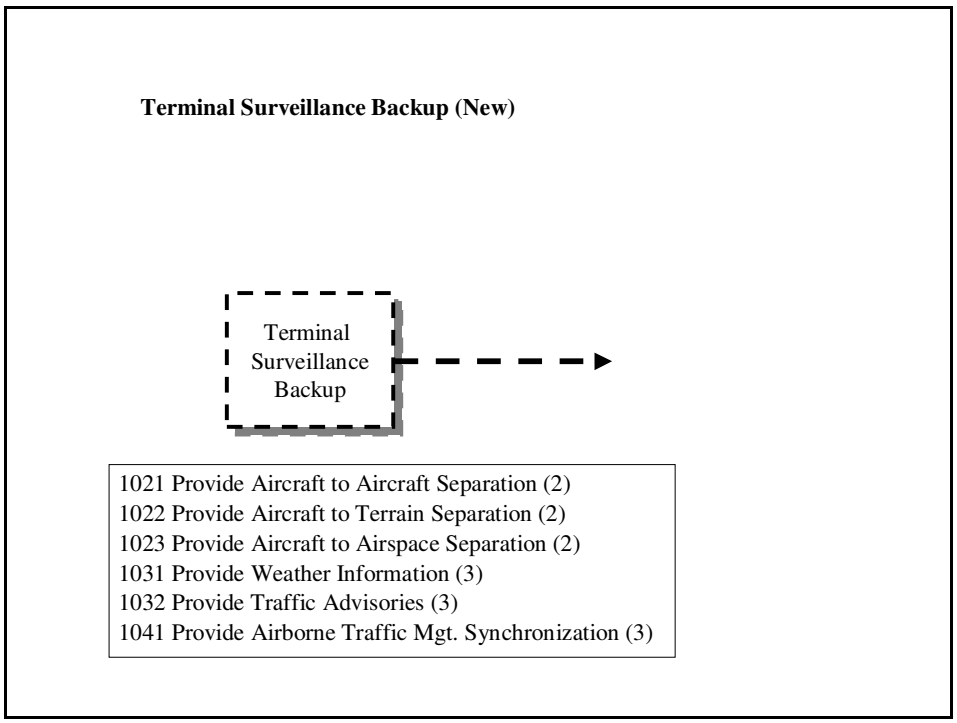

FIGURE E - 43: Terminal Surveillance Backup (New)

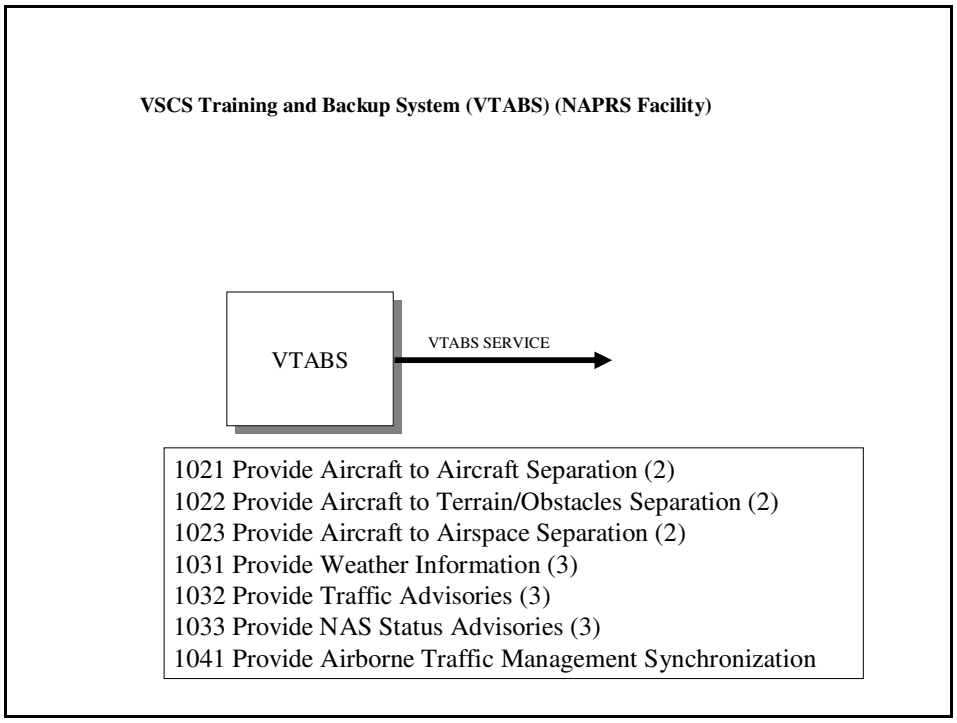

FIGURE E - 44: VSCS Training and Backup System (VTABS) (NAPRS Facility)

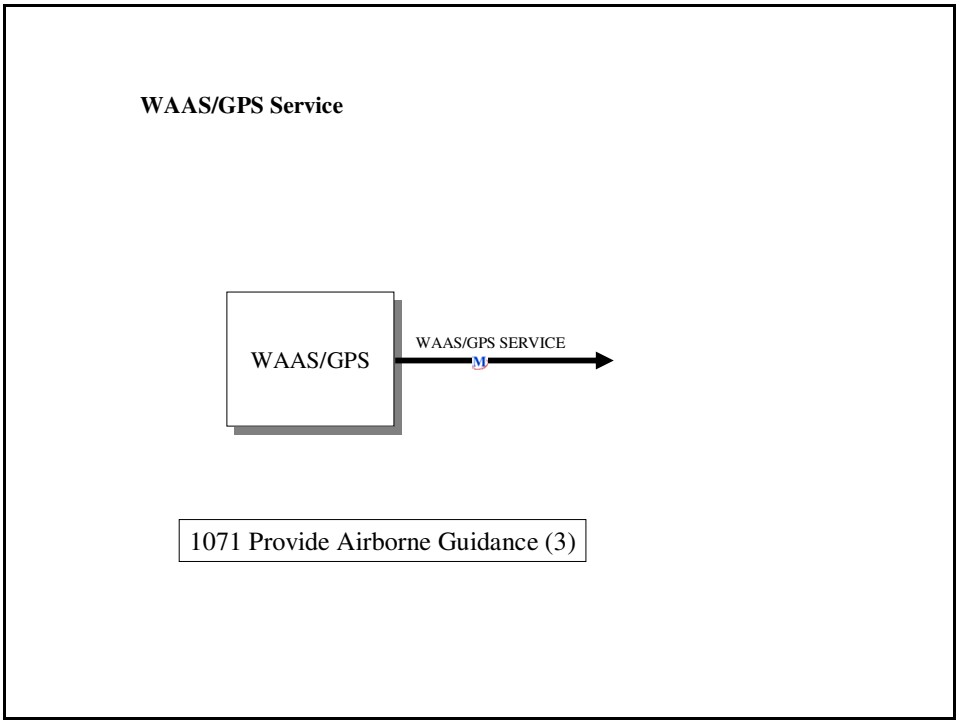

FIGURE E - 45: WAAS/GPS S Service

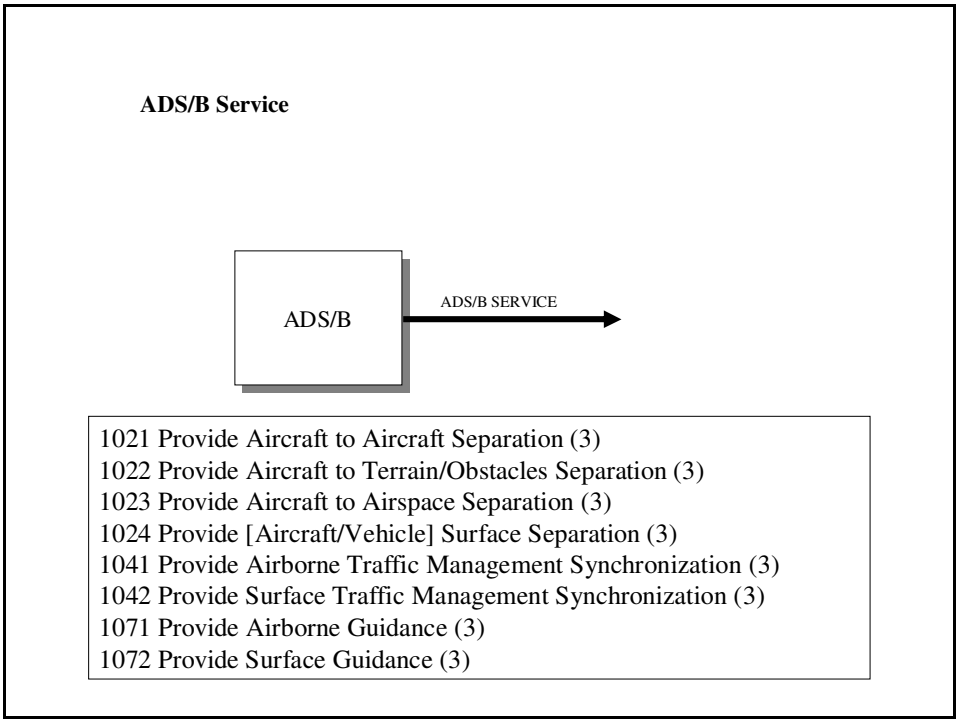

FIGURE E - 46: ADS/B Service

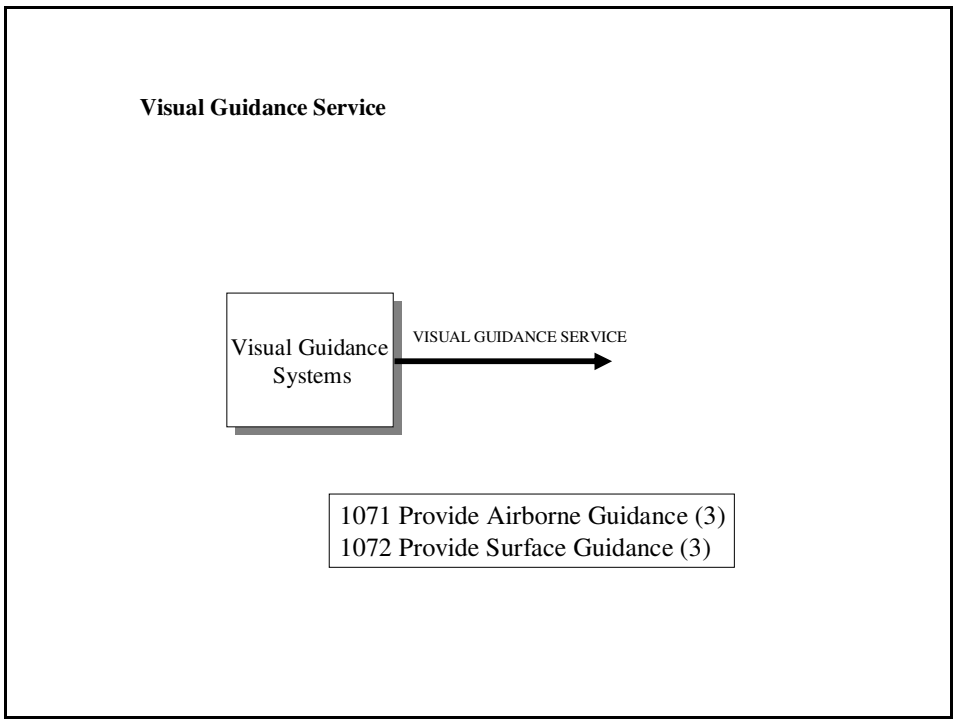

FIGURE E - 47: Visual Guidance Service

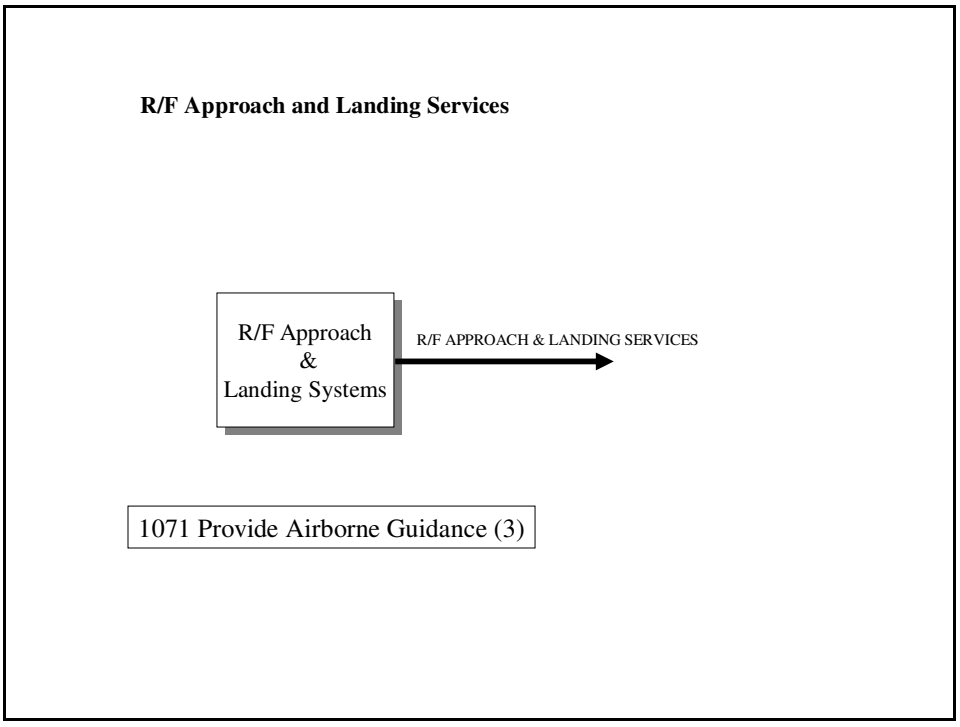

FIGURE E - 48: R/F Approach and Landing Services

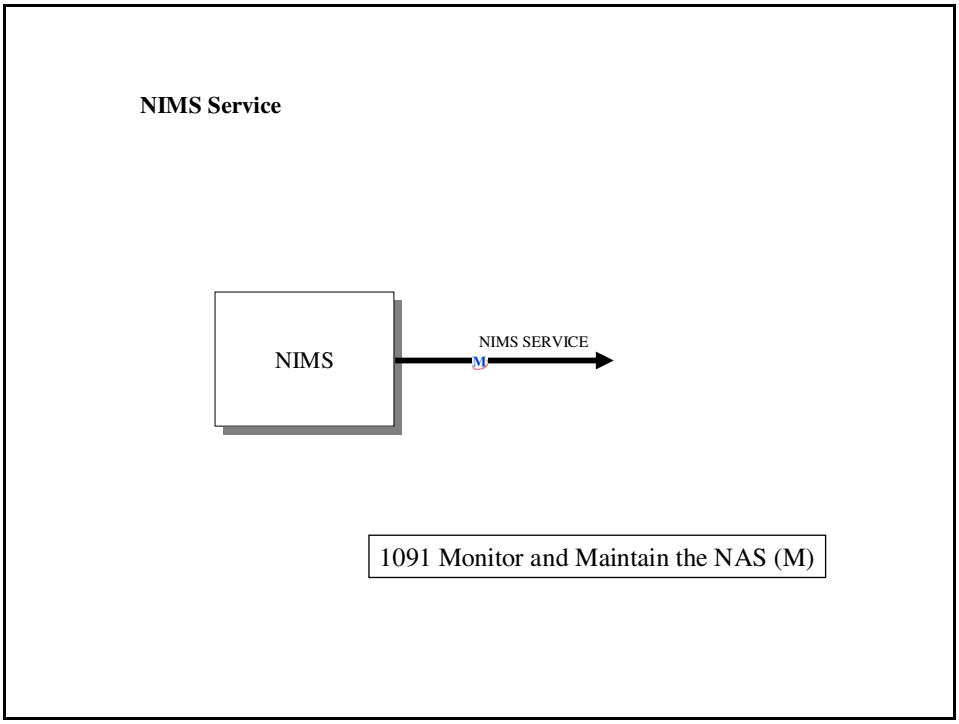

FIGURE E - 49: NIMS Service

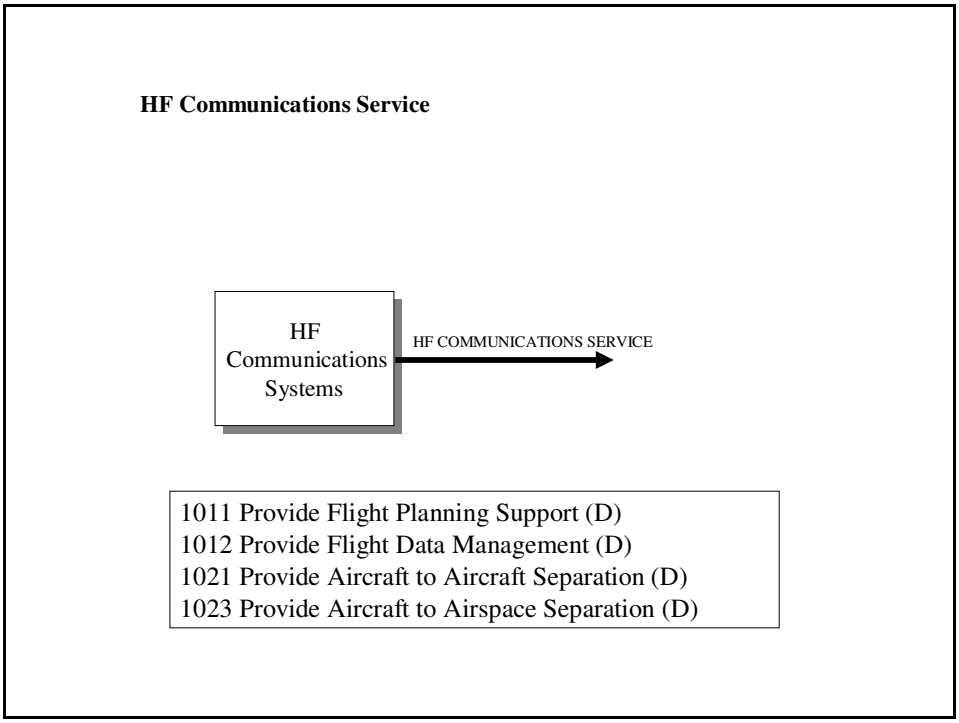

FIGURE E - 50: HF Communications Service

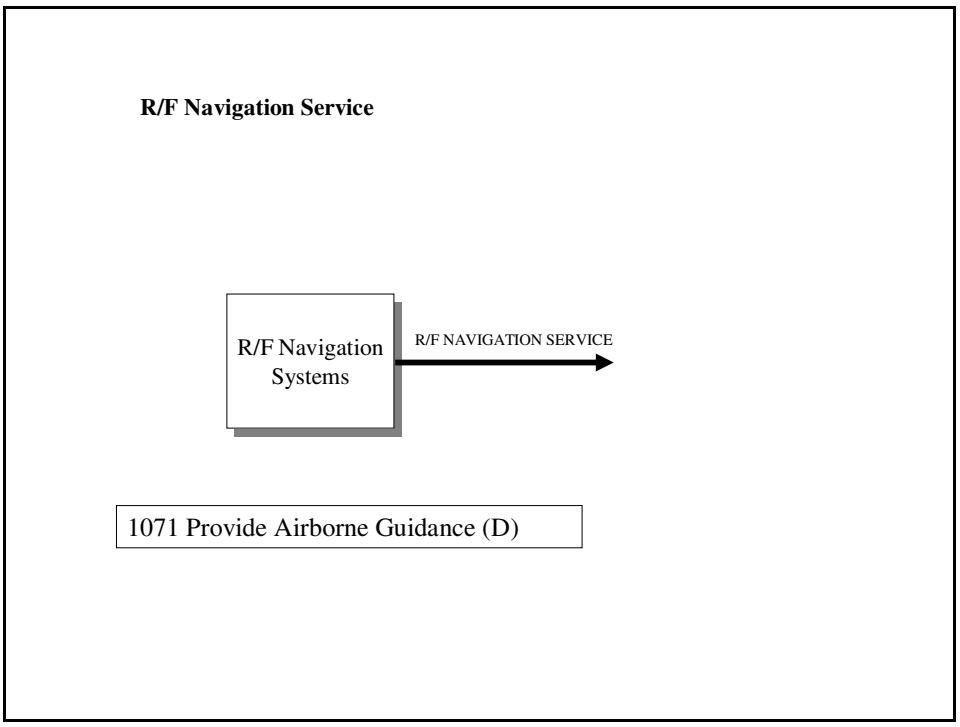

FIGURE E - 51: R/F Navigation Service

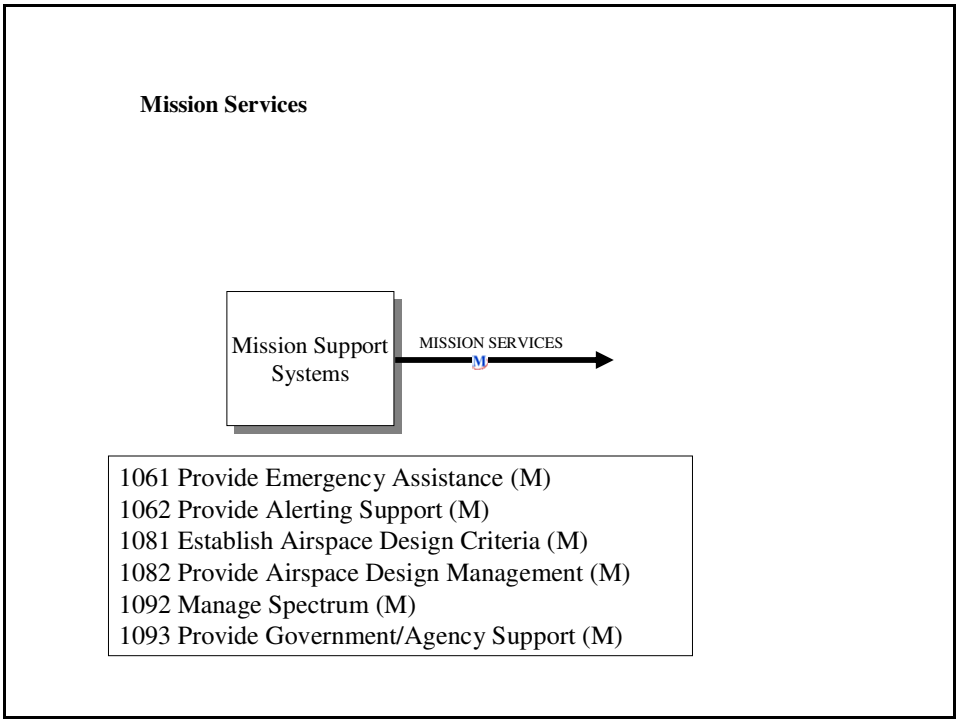

FIGURE E - 52: Mission Services

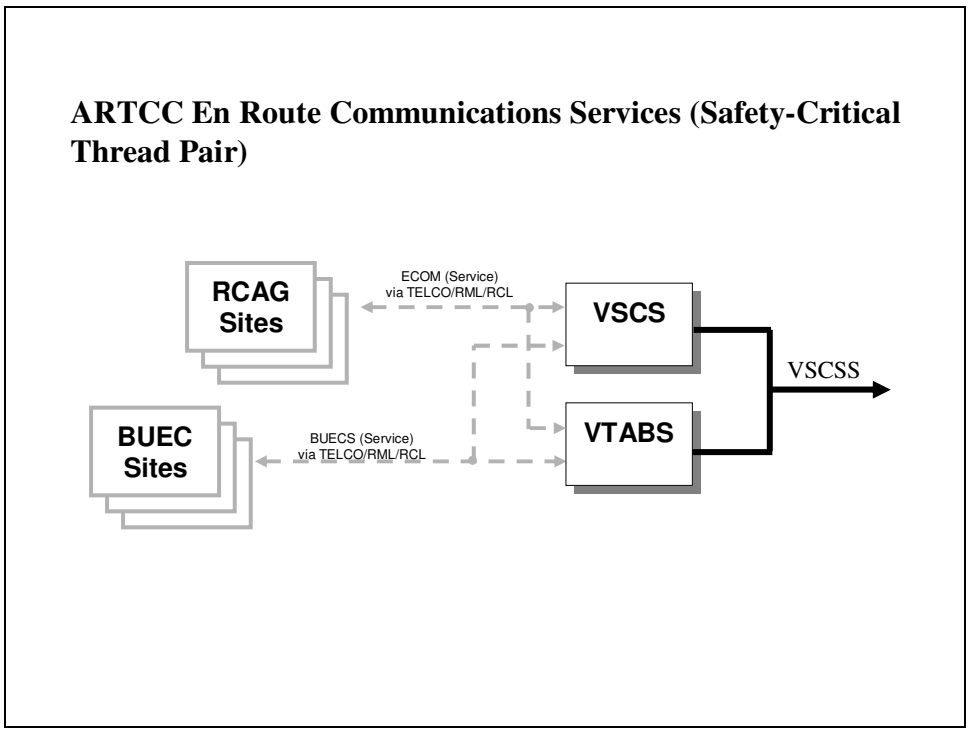

FIGURE E - 53: Safety-Critical En Route Communications Service Thread Pair

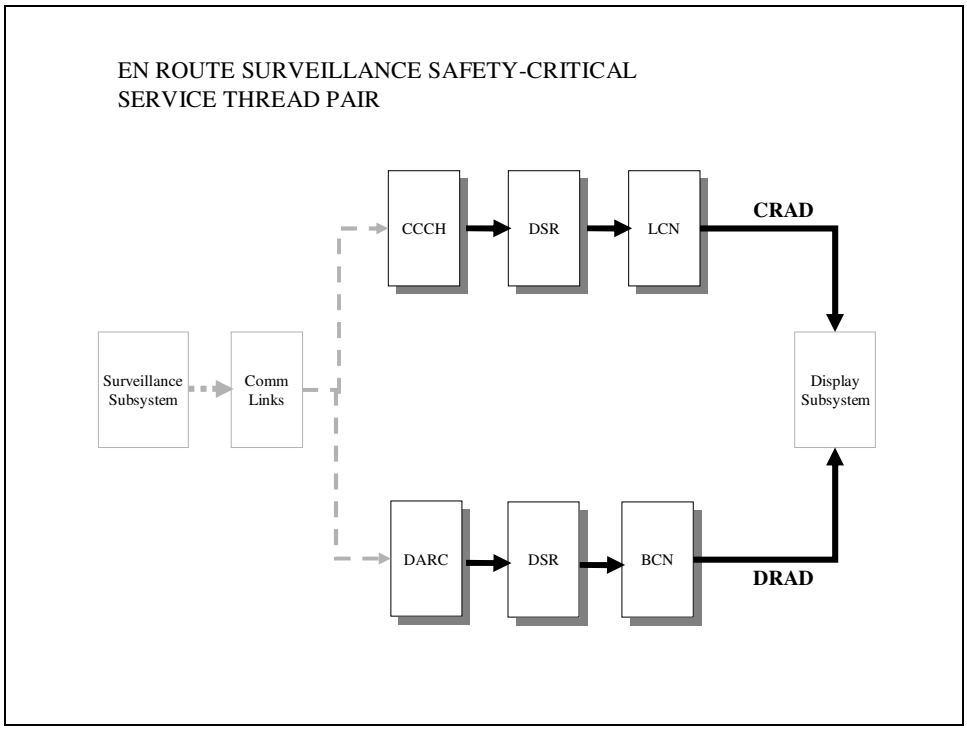

FIGURE E - 54: Safety-Critical En Route Surveillance Service Thread Pair

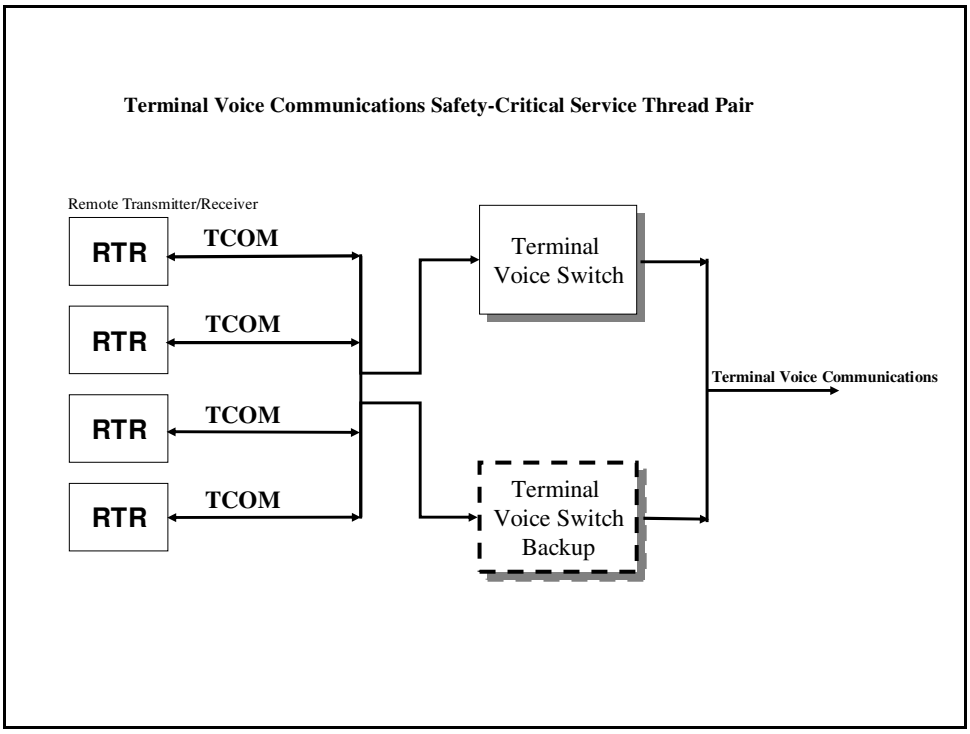

FIGURE E - 55: Safety-Critical Terminal Voice Communications Service Thread Pair

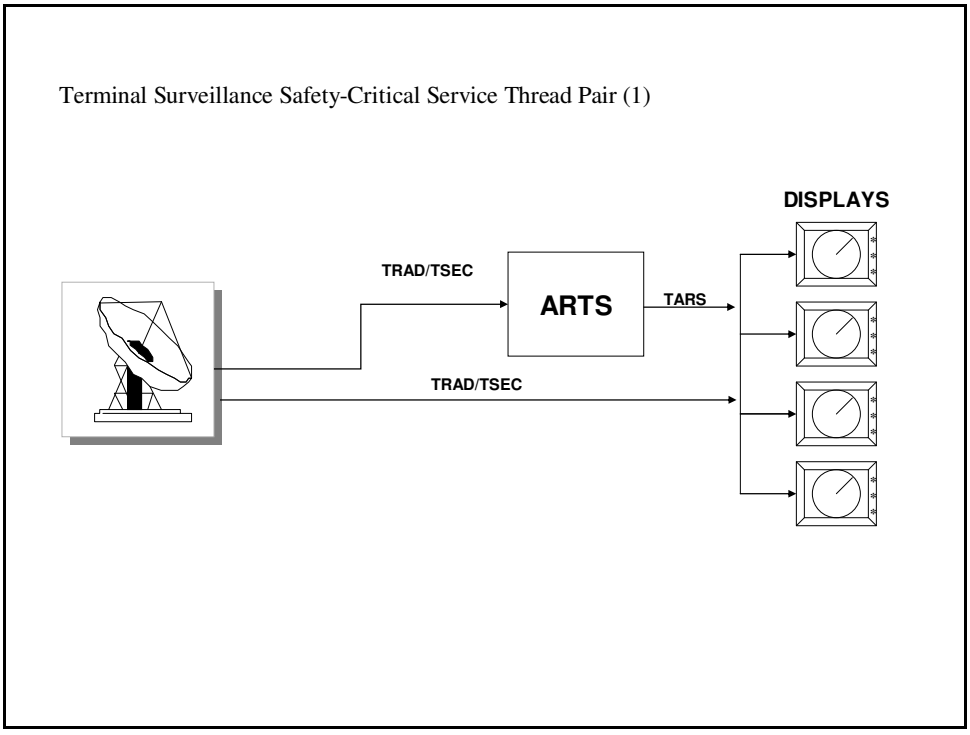

FIGURE E - 56: Safety-Critical Terminal Surveillance Service Thread Pair (1)

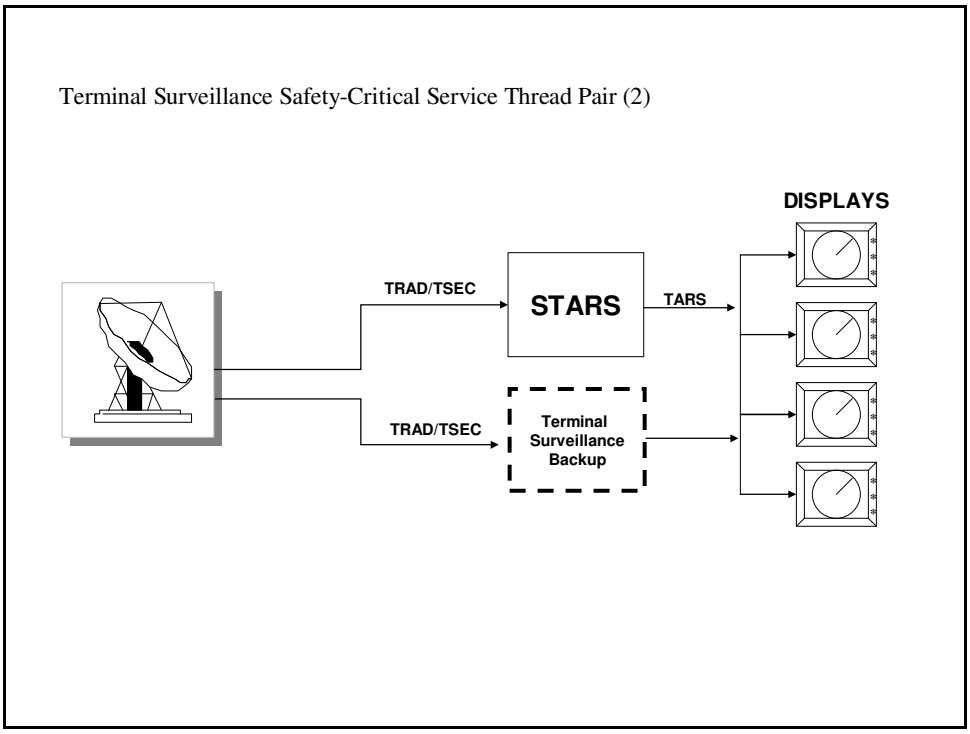

FIGURE E - 57: Safety-Critical Terminal Surveillance Service Thread Pair (2)

Appendix F LIST OF ACRONYMS

Acronym	Full Form
AF	Airways Facilities
AFSS	Automated Flight Service Station
ARSR	Air Route Surveillance Radar
ARTCC	Air Route Traffic Control Center
ASR	Airport Surveillance Radar
ATC	Air Traffic Control
ATCT	Airport Traffic Control Tower
AWP	Aviation Weather Processor
CCC	Central Computer Complex
CDR	Critical Design Review
CFAD	Composite Flight Data Processing
CHI	Computer/Human Interface
COTS	Commercial Off-the-Shelf
CPCI	Computer Program Configuration Item
CRAD	Composite Radar Data Processing
CWG	Communications Working Group
DID	Data Item Description
DR&A	Data Reduction and Analysis
DRACAS	Data Reporting, Analysis, and Corrective Action System
DRAD	DARC Radar Data Processing
FCA	Functional Configuration Audit
FMEA	Failure Mode and Effects Analysis
FMEA	Failure Modes and Effects Analysis
FMECA	Failure Mode, Effects and Criticality Analysis
FRACAS	Failure Reporting, Analysis, and Corrective Action System

FSDPS	Flight Service Data Processing System
HVAC	Heating, Ventilating, and Air Conditioning
IFPP	Information for Proposal Preparation
ILS	Instrument Landing System
LRU	Lowest Replaceable Unit
M&C	Monitor and Control
MDT	Mean Down Time
MSI	Maintenance Significant Items
MTBF	Mean Time Between Failures
MTBO	Mean Time Between Outages
MTTR	Mean Time to Repair
MTTRS	Mean Time to Restore Service
NADIN	National Airspace Data Interchange Network
NAPRS	National Airspace Reporting System
NAS	National Airspace System
NASPAS	National Airspace System Performance Analysis System
NDI	Non-Developmental Item
PCA	Physical Configuration Audit
PDR	Preliminary Design Review
PTR	Program Trouble Report
RCAG	Remote Communications Air-Ground
RMA	reliability, maintainability and availability
RTR	Remote Transmitter/Receiver
SAR	System Analysis and Recording
SDR	System Design Review
SEM	System Engineering Manual
SLS	System-Level Specification
SOW	Statement of Work

SRR	System Requirements Review
STLSC	Service Thread Loss Severity Category
TIM	Technical Interchange Meeting
TRACON	Terminal Radar Approach Control
VOR	VHF Omnidirectional Range
WJHTC	William J. Hughes Technical Center

1. Reliability, Maintainability, and Availability Draft of SR 1000 as of October 16, 2007

2. This section identifies the NAS level RMA requirements. FAA RMA Handbook – 006A allocates the RMA requirements to the Services and Capabilities which are supported by one or more strings of systems called Service Threads. Service Threads bridge the gap between un-allocated functional requirements and the specifications for systems that support them.

2.1. Availability Requirements

2.1.1. NAS Capability Availability

2.1.2. Critical NAS Capabilities shall have a target operational availability equal to or greater than to .99999.

2.1.3. Essential NAS Capabilities shall have a target operational availability equal to or greater than to .999.

2.1.4. Routine NAS Capabilities shall have a target operational availability equal to or greater than to .99.

2.1.5. Critical NAS Capabilities shall be supported by two or more independent Service threads.

2.1.6. Service Thread Inherent Availability

2.1.7. Essential Service threads shall have a target inherent availability greater than .9999.

2.1.8. Efficiency-critical Service threads shall have a target inherent availability equal to or greater than to .99999

2.1.9. Safety-critical Capability Availability

2.1.10. Safety-critical capabilities shall be supported by two or more independent efficiency-critical service threads with each thread having a target inherent availability equal to or greater than to .99999.

2.1.11. Manual means shall be provided to select between the independent service threads providing safety-critical capabilities.

2.1.12. Remote/Distributed Service Thread Availability

2.1.13. Remote/Distributed Service threads shall have a target inherent availability equal to or greater than to the inherent availability of the supported NAS Capabilities.

2.1.14. Power Availability

2.1.15. Power for ATC operations at Level 11-12 terminal facilities shall have an inherent availability equal to or greater than to .999998.

2.1.16. Power for ATC operations at Level 8-10 terminal facilities shall have an inherent availability equal to or greater than to .999998.

2.1.17. Power for ATC operations at Level 6-7 terminal facilities shall have an inherent availability equal to or greater than to.9998.

2.1.18. Power for ATC operations at Level 4-5 terminal facilities shall have an inherent availability equal to or greater than to .9998.

2.1.19. Power for ATC operations at Level 2-3 terminal facilities shall have an inherent availability equal to or greater than to of .9998.

2.1.20. Power for ATC operations at Level 1 terminal facilities shall have an inherent availability of .9998 or greater.

2.1.21. Power for ATC operations at en route facilities shall have an inherent availability equal to or greater than to .999998.

2.1.22. Power for remote communications facilities shall have an inherent availability equal to or greater than to .99.
2.1.23. Power for remote surveillance facilities shall have an inherent availability equal to or greater than to .99.
2.1.24. Power for remote navigation facilities shall have an inherent availability equal to or greater than to .99.

2.2. Maintainability Requirements

2.2.1. The Mean Time to Restore (MTTR) for Information System thread components shall be less than or equal to 0.5 hours.
2.2.2. The Mean Time to Restore (MTTR) for Remote/Distributed system thread components shall be less than or equal to 0.5 hours.
2.2.3. The Mean Time to Restore (MTTR) for power system components shall be less than or equal to 0.5 hours.

2.3. Reliability Requirements

2.3.1. The Mean Time Between Failures (MTBF) for Information System service threads with automatic recovery requirements and whose recovery time is less than the Automatic Recovery Time shall be equal to or greater thanl to 300 hours.
2.3.2. The MTBF for Information System service threads with automatic recovery requirements and whose recovery time is greater or equal to the Automatic Recovery Time shall be equal to or greater thanl to 50,000 hours.
2.3.3. The MTBF for Information System service threads that have no automatic recovery requirement shall be equal to or greater thanl to 5,000 hours.

Definitions

Availability – The probability that a system or constituent piece may be operational during any randomly selected instant of time or, alternatively, the fraction of the total available operating time that the systems or constituent piece is operational.

Inherent Availability – The theoretical availability of a system or constituent piece.

Operational Availability – The availability including *all* sources of downtime, both scheduled and unscheduled.

Information Systems – Information systems receive inputs from one or more external inputs, process that information, and prepare it for output on one or more output devices.

Independent Service Threads – Threads composed of separate system components. Such threads may share a single power source, provided that power source is designed to minimize failures that could cause both service threads to fail. Such threads may share displays provided that such adequate redundant displays are provided to permit the specialist to relocate to an alternate display in the event of a display failure.

MTBF – Mean Time Between Failure – The mean number of life units during which all parts of the system or constituent pieces perform within their specified limits, during a particular measurement interval under stated conditions.

MTTR – Mean Time to Restore – The total elapsed time from initial failure to resumption of operation.

NAS Service/Capability Criticality – The severity of the impact of the loss of that Service/Capability has on the safe and efficient operation and control of aircraft.

- Critical – Loss of this Service/Capability would raise the risk associated with providing safe and efficient local NAS operations to an unacceptable level.
- Essential – Loss of this Service/Capability would significantly raise the risk associated with providing safe and efficient local NAS operations.
- Routine – Loss of this Service/Capability would have a minor impact on the risk associated with providing safe and efficient local NAS operations.

Service Threads – Service threads are strings of systems that support one or more service/capabilities to a user/specialist.

Service Thread Loss Severity Category - The severity of impact of the loss of a service thread on the safe and efficient operation and control of aircraft of having to transition from that Service Thread to another Service Thread or to a reduced level of capacity operations:

a) **Safety-critical:** Service Thread loss would present an unacceptable safety hazard during the transition to reduced capacity operations.

b) **Efficiency-critical**– Service Thread loss could be accommodated by reducing capacity without compromising safety, but the reduced capacity operation has the potential for system-wide impact on NAS efficiency.

c) **Essential–** Service Thread loss can be accommodated without compromising safety and with only localized impact on NAS efficiency.

Target Operational Availability – The desired operational availability associated with a given NAS Service/Capability Criticality.

Remote/Distributed Service Thread – Service threads composed of systems or components which are located at remote sites such as remote communications, inter-facility data communications and navigation sites, as well as distributed subsystems such as display terminals that may be located within a major facility. Failures of single or multiple component elements, may degrade performance, but generally do not result in the total loss of the Service Threads capability.