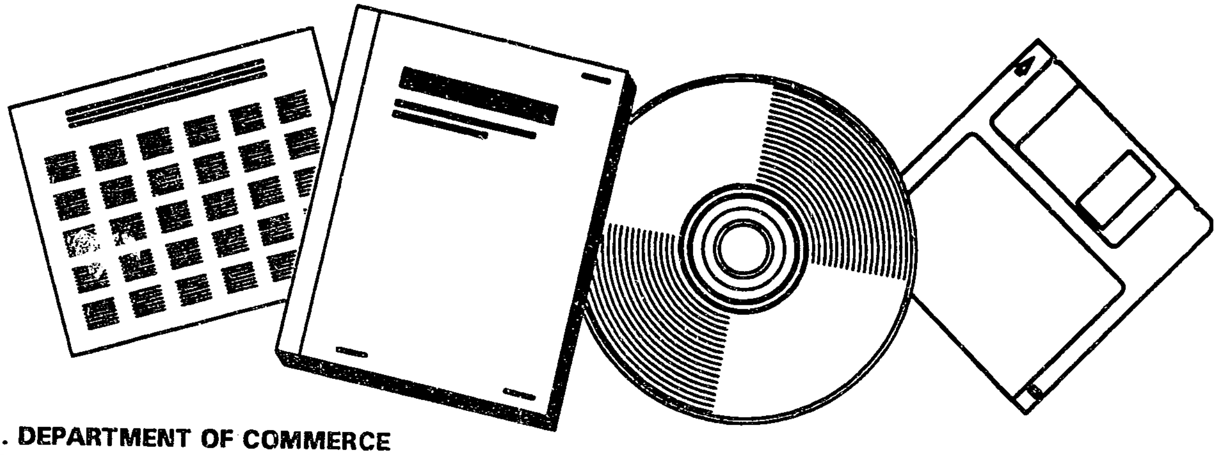
PB96-109772

Information is our business.

SAFETY OF HIGH SPEED GROUND TRANSPORTATION SYSTEMS: ANALYTICAL METHODOLOGY FOR SAFETY VALIDATION OF COMPUTER CONTROLLED SUBSYSTEMS. VOLUME 1. STATE-OF-THE-ART AND ASSESSMENT OF SAFETY VERIFICATION/VALIDATION METHODOLOGIES

(U.S.) JOHN A. VOLPE NATIONAL TRANSPORTATION SYSTEMS CENTER
CAMBRIDGE, MA

SEP 95



U.S. DEPARTMENT OF COMMERCE
National Technical Information Service

U. S. Department of Transportation

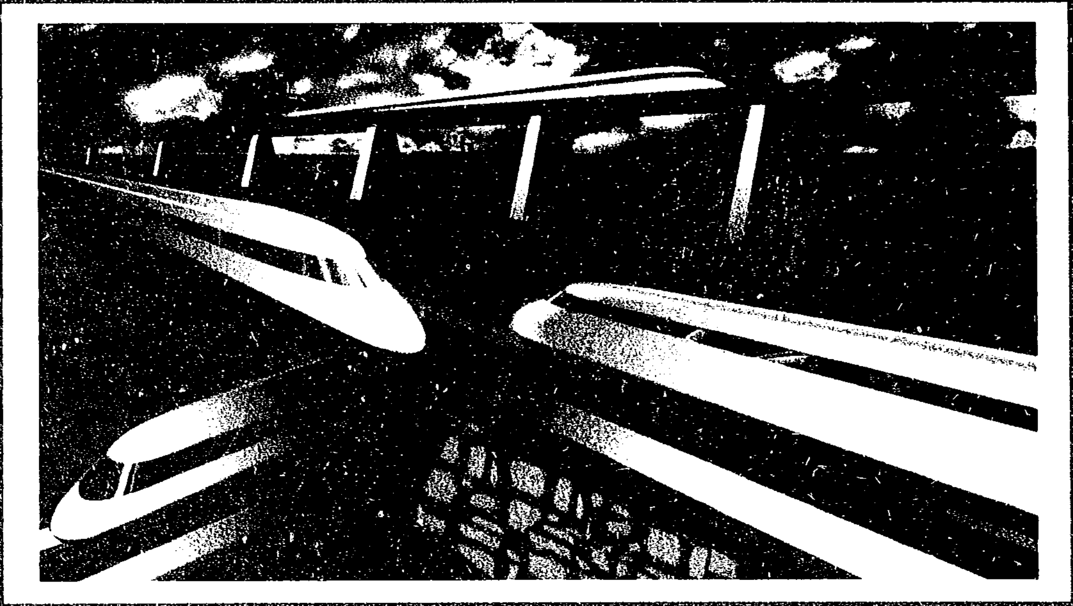
Federal Railroad Administration

Safety of High Speed Ground Transportation Systems

PB96-109772

Office of Research
and Development
Washington, D.C. 20590

Analytical [illegible]
of [illegible]



DOT/FRA/ORD-95/10.1
DOT-VNTSC-FRA-95-8.I

Final Report
September 1995

This document is available to the public through the National Technical Information Service, Springfield, VA 22161

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services. Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA
ent and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

	2. REPORT DATE September 1995	3. REPORT TYPE AND DATES COVERED Final - April 1994

4. TITLE AND SUBTITLE Safety of High Speed Ground Transportation Systems: Analytical Methodology for Safety Validation of Computer Controlled Subsystems Volume I: State-of-the-Art and Assessment of Safety Verification/Validation Methodologies	5. FUNDING NUMBERS RR593/R5019
6. AUTHOR(S) Jonathan F. Luedeke*	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Battelle 505 King Avenue Columbus, OH 43201-2693	8. PERFORMING ORGANIZATION REPORT NUMBER DOT-VNTSC-FRA-95-8.I
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) U.S. Department of Transportation Federal Railroad Administration Office of Research and Development Washington, DC 20590	10. SPONSORING/MONITORING AGENCY REPORT NUMBER DOT/FRA/ORD-95/10.1

11. SUPPLEMENTARY NOTES
*under contract to:

U.S. Department of Transportation
Research and Special Programs Administration
Volpe National Transportation Systems Center
Kendall Square, Cambridge, MA 02142

12a. DISTRIBUTION/AVAILABILITY STATEMENT This document is available to the public through the National Technical Information Service, Springfield, VA 22161	12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This report describes the development of a methodology designed to assure that a sufficiently high level of safety is achieved and maintained in computer-based systems which perform safety critical functions in high-speed rail or magnetic levitation transportation systems. This report consists of two volumes. This, the first presents a glossary of relevant computer technology terminology to assure consistency of use and understanding. A state-of-the-art review of safety verification and validation processes worldwide is presented. Following the review, these processes are assessed relative to their degree of assured safety as well as their potential applicability to safety critical systems in US rail transportation systems.

The second volume builds upon the information developed in the first volume and describes a methodology which has been developed specifically for application to computer-controlled systems used in railroad applications in the United States.

14. SUBJECT TERMS verification, validation, software, hardware, methodology, safety, safety standards, high-speed rail, magnetic levitation, high-speed guided ground transportation system	15. NUMBER OF PAGES 224
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT

PREFACE

The Federal Railroad Administration (FRA) is responsible for assuming the safety of high-speed rail and magnetic levitation systems deployed in this country. A primary concern of the FRA is the proper use of computer technology in the implementation of safety critical functions, such as signalling and train control, in newer high-speed systems as well as in conventional rail systems. This report describes the development of a methodology designed to assure that a sufficiently high level of safety is achieved and maintained in these computer-based systems. Adequate safety is necessary whether the systems are used in new applications or are used to replace or enhance existing equipment. This report comprises the first of two volumes relative to the development of this methodology.

In this report, a glossary of terms has been developed to ensure consistency and understanding. Next, a description of the state-of-the-art in safety verification and validation methodologies worldwide is presented. Finally, an assessment of these methodologies from the standpoint of their applicability and level of assured safety is conducted.

The second volume builds upon this work and describes a methodology which has been developed specifically for application to computer-controlled systems used in railroad applications in the United States.

This report was prepared in support of the United States Department of Transportation, FRA, Office of Research and Development.

TABLE OF CONTENTS

Section **Page**

TABLE OF CONTENTS (cont.)

TABLE OF CONTENTS (cont.)

TABLE OF CONTENTS (cont.)

Section **Page**

LIST OF FIGURES

Figure **Page**

LIST OF TABLES

Table **Page**

1. INTRODUCTION

The Federal Railroad Administration (FRA) is currently responsible for assuring the safety of conventional rail, high-speed rail and maglev systems deployed in this country. One of the FRA's primary concerns is the proper use of computer technology in the implementation of safety critical functions in newer high-speed systems as well as in conventional rail systems. Existing federal regulations governing signalling and train control systems may need to be revised to adequately address the various issues associated with the utilization of this new technology.

The Volpe National Transportation Systems Center (Volpe Center) is assisting the FRA in identifying and addressing many of the pertinent safety issues. The primary interest in this overall program, conducted for the Volpe Center in support of the FRA, is the development of a methodology to assure that a sufficiently high level of safety is achieved and maintained in these computer-based systems. Adequate safety is necessary whether the systems are used in new applications or are used to replace or enhance existing signalling/train control equipment.

This document is the Final Report for the Base Task (or first of two major tasks) of the program relative to the development of this methodology. The report describes work performed and results obtained on three major activities or items of work. The first (i.e., Item 1) involved the definition of terminology and acronyms relevant to the safety verification and validation of computer-controlled subsystems used in railroad and other fixed guideway applications including high-speed rail and maglev. The second (i.e., Item 2) involved a description of the state-of-the-art in safety verification and validation methodologies and associated standards in computer-based systems worldwide. The third (i.e., Item 3) involved an assessment of the methodologies from the standpoint of their applicability and level of assured safety.

Results of this work will serve as a basis for the remaining task (i.e., Option Task) of this program in which a specific methodology will be developed and recommended to the FRA.

1.1 BACKGROUND

The evolution in the implementations of safety critical systems in the railroad industry from simple vital relays to more complex computer-based configurations has raised many issues among users as well as the FRA. Foremost among these issues is the need to assure similar or improved levels of safety to those currently provided by conventional fail-safe technology. This concern is heightened in newer high-speed rail and maglev systems which operate or are being designed to operate at considerably higher speeds and levels of automation than conventional rail systems. Computers are playing an increasing role in the safety critical functions in these newer systems such as in train location determination, switch/route control (interlocking), control of braking/propulsion to ensure safe speed and headway, and communications among the trains, wayside and central elements.

end of development), it was necessary to investigate the entire development life cycle of a system.

As was observed during this study, there is a lack of common usage in the various methodologies and standards addressed relative to the terms "verification" and "validation" as well as "safety verification" and "safety validation." In many of the methodologies/standards reviewed, the terms verification and validation are used in a similar manner to that conveyed in the Institute of Electrical and Electronics Engineers (IEEF) document "Standard Glossary of Software Engineering Terminology," IEEE Std. 610.12-1990. In that document, which pertains specifically to software, the terms verification and validation are defined as follows:

- Verification -- The process of determining whether or not the products of a given phase of the (software) development life cycle fulfill the requirements established during the previous phase.

- Validation -- The process of evaluating (software) at the end of the software development process to ensure compliance with software requirements.

It should be noted that the above definitions are not specifically directed to safety or safety requirements. Rather, they apply to software requirements in general. Many of the methodologies reviewed in this study use similar definitions when dealing with safety requirements for software, and also extend these definitions to address system and even hardware safety requirements. In these instances, the methodologies use terms such as safety verification, safety validation, system validation and even software and hardware verification and validation. Thus, many variations exist, and in order to determine the actual definition of one of these terms in a given methodology, it is necessary to understand the context in which it is used.

In general, for purposes of this study, a safety validation is considered to be a process or set of activities performed on a system, software or hardware element to demonstrate compliance with safety requirements. It typically is performed on a completed system or hardware or software element. A safety verification can be synonymous with safety validation, but it is considered in this program as an incremental confidence building activity or process performed following a given phase of system, software, or hardware development to determine compliance with safety requirements established for that phase.

Also, as a point of clarification, the phrase "safety V&V" is used throughout this report, often in place of the phrase "safety verification and validation." The two phrases are synonymous and refer to any or all safety verification and/or safety validation activities that may be associated with a given methodology.

It should be emphasized that this study was directed to the safety verification and validation aspects of the various overall safety assurance processes used by different organizations and/or described in the various standards or guideline documents. Many of the safety assurance processes, standards and guideline documents investigated address other aspects and requirements relative to such areas as design/development, quality assurance, documentation, safety management and others. While these other aspects do play various roles in helping to ensure overall system safety, they could not be treated in the same manner or to the same extent in this assessment as safety verifications/validations. This is due to the scope of work of this project and the extensiveness of other standards and guidelines which address other aspects instead of, or in addition to, safety-related verifications and validations.

2. APPROACH

As indicated, this program was separated into two major tasks (i.e., Base Task and Option Task). This section provides brief overviews of the work performed in the Base Task (to which this report is directed) and the work to be performed in the follow-on Option Task.

2.1 BASE TASK APPROACH

Work performed in the Base Task was separated into three major items of work as identified below:

- Item 1 - Glossary of Terms – Development of a glossary of terms and acronyms which may be encountered throughout the course of this program.
- Item 2 - State-of-the-Art in Safety Verification/Validation Methodologies Review of the state-of-the-art in safety verification and validation methodologies and standards as developed by railway equipment suppliers; regulatory bodies such as the Nuclear Regulatory Commission (NRC); and other organizations such as the IEEE, Department of Defense (DOD), International Union of Railways (UIC) and the Federal Aviation Administration (FAA).
- Item 3 - Assessment of Existing Safety Verification/Validation Methodologies – Assessment of the methodologies described in Item 2 from the standpoint of applicability (to railroad and other fixed guideway technology) and level of assured safety.

Interim reports were generated for each item of work, and all results were assimilated into this comprehensive final report.

A fourth activity (i.e., Item 4 - Techno-Economic Feasibility Study) was originally planned for this Base Task. However, since a single "best" existing methodology is not being recommended at this time (for reasons discussed later), the feasibility study will be conducted later in this program.

The nature of the work performed on each of the Base Task activities is described below.

2.1.1 Item 1 Approach - Glossary of Terms

Item 1 involved the development of a glossary of terms pertaining to the safety verification and validation of computer-controlled subsystems used in railroad and fixed guideway applications. Work was initiated by establishing a list of relevant and appropriate terms and acronyms pertaining to several topic areas. Areas of interest included safety, computer systems, software and software engineering, verification and validation, signalling and train control, and implementations of systems/equipment to which the methodology (to be developed later in this project) will be applied.

Over 25 documents containing definitions of terms in the above areas were identified and obtained. This included documents from the Institute of Electrical and Electronic Engineers (IEEE), the National Computer Systems Laboratory (NCSL), the Association of American Railroads (AAR), the American Public Transit Association (APTA), the Department of Defense (DOD), the Volpe Center and others.

Definitions considered to be the most relevant for this program were extracted from the literature. Although multiple definitions were found for numerous terms, every attempt was made to select the most clear, concise and appropriate definitions given the nature of this program and the fact that the glossary will be used by a variety of personnel with different skills and backgrounds. Specific reference sources were cited for the definitions selected.

The glossary is provided in Appendix A together with all associated reference sources.

2.1.2 Item 2 Approach - State-of-the-Art In Safety Verification/Validation Methodologies

Item 2 involved the identification and description of safety verification and validation (V&V) methodologies being utilized by various railway, regulatory bodies and other organizations worldwide to assess the safety of computer-based systems/equipment. This work included the identification and description of various safety related standards/guidelines which were required either in part or in full by the methodologies themselves. As dictated by the scope of work, emphasis was on the railroad industry.

A list of 22 organizations to be addressed in the study was established and jointly agreed upon by the Volpe Center and Battelle at the project's initiation. Included were railway suppliers and authorities, regulatory bodies and other organizations from North America, Europe and Japan. As the study progressed and information was obtained, (six) additional organizations were added to this list due to their unique safety V&V processes/standards.

In order to obtain information on the various safety V&V methodologies/standards used, appropriate personnel involved with each organization were identified and contacted, after which follow-up letters were sent to outline the information of interest. It was quickly observed that, in most instances, a single document which described the safety V&V process used by a specific firm did not exist. Rather, the process typically involved multiple internal documents (some of which were proprietary) and/or existing/draft safety standards and other nonmandatory guidelines. Thus, it was usually necessary to obtain multiple documents for each organization from (usually) several different sources both within and external to the organization.

Following numerous discussions and a review of all literature received, summary descriptions were prepared of the safety V&V methodologies/standards used or developed by the different organizations. The intent in each of these descriptions was to summarize the following: 1) the role of the organization in setting standards, conducting safety V&V and/or obtaining approval/certification of systems/equipment, 2) the identification of existing standard/methodology documentation utilized or developed, and 3) the nature/content of the safety V&V process itself — what activities are performed, why they are performed, when in the product development they are applied, and who performs them.

2.1.3 Item 3 Approach - Assessment of Safety Verification/Validation Methodologies

Item 3 involved an assessment of the safety verification and validation methodologies developed by various organizations to help ensure the safety of computer-based systems/ equipment. The organizations addressed include those addressed in the Item 2 activity plus several others for which unique safety-related methodologies or standard/guideline documents (which include safety V&V aspects) were identified.

The assessment was conducted in two parts from two major standpoints: 1) applicability to railroad and other fixed guideway equipment, and 2) level of assured safety. First, an initial assessment was performed in order to select a lesser number of the most promising methodologies for further and more detailed review. Criteria used in this initial assessment were directed to some general aspects as well as the potential applicability of the methodologies. Second, a more detailed assessment was conducted in which the selected methodologies were subjected to other criteria which were heavily directed to the level of assured safety if the methodologies were to be applied. Attributes and limitations of each methodology were identified, and an overall summary was prepared.

2.2 OPTION TASK APPROACH

Further efforts in this overall program (in the Option Task) will be separated into the following activities:

- Development/Recommendation of a specific methodology (based upon the results of the Base Task) and compliance ensurance process for FRA's consideration.

- Development of a training program for FRA personnel on ensuring compliance with the developed methodology.

- Conduct of a techno-economic feasibility study of the recommended methodology, and

- Assessment of human factor issues relative to computer automation and operators' interfaces.

3. SAFETY VERIFICATION AND VALIDATION METHODOLOGIES

This section presents descriptions of the safety verification and/or validation methodologies being utilized or developed by the following organizations/industries for application to safety critical computer-based systems:

- General Railway Signal (GRS)
- Union Switch & Signal (US&S)
- Harmon Electronics
- ALCATEL - Canada
- ATCS (Advanced Train Control System)
- British Rail
- International Union of Railways (UIC) - France/European Rail Research Institute (ERRI) - Netherlands
- TÜV Rheinland - Germany
- German Federal Railway (DB) - Germany
- ABB Signal AB - Sweden
- Siemens AG - Germany
- Matra Transport* - France
- Sasib - Italy
- French National Railway (SNCF) - France
- Ministry of Defence* - United Kingdom
- Institution of Railway Signal Engineers (IRSE)*
- Railway Technical Research Institute (RTRI)* - Japan
- East Japan Railways - Japan
- Nippon Signal - Japan
- Hitachi** - Japan
- International Electrotechnical Commission (IEC)* - Switzerland
- IEEE
- Department of Defense (DOD)
- Federal Aviation Administration (FAA)
- National Aeronautics and Space Administration (NASA)
- Nuclear Regulatory Commission (NRC)
- Medical Industry
- Underwriters Laboratory*

* In addition to those originally agreed upon.

** No information received/available.

TABLE 3-1. MAJOR SAFETY VERIFICATION/VALIDATION STANDARDS/GUIDELINES FOR COMPUTER-BASED SYSTEMS

Issuing Organization	Document Number	Title	Intended Application
Railway Association of Canada/ Association of American Railroads	ATCS Specification 140	Recommended Practices for Safety and Systems Assurance	Railroad (ATCS)
	ATCS Specification 130	Recommended Practices for Software Quality Assurance	
Transport Canada	TP 10770 E	ATCS System Safety Validation Programs	Railroad (ATCS)
RIA	Technical Specification No. 23	Safety-Related Software for Railway Signalling	Railroad
UIC	738R	Processing and Transmission of Safety Information	Railroad
UIC/ORE	A155/RP11	Proof of Safety of Computer-Based Safety Systems	Railroad
	A155.1/RP8	On Proving the Safety of Transmission Systems	
TÜV Rheinland	SBT 90.01/00/E	Guidelines for the Assessment of Safety-Relevant Computer Systems in Railroad Technology	Railroad
German Federal Railway	Mü 8004	Principles of Technical Approval for Signalling and Communications Technology	Railroad
CENELEC	CLC/TC9X/ SC9XA/WGA1 (Draft)	Railway Applications: Software for Railway Control and Protection Systems	Railroad
	CLC/TC9X/ SC9XA/WGA2 (Draft)	Railway Applications: Safety-Related Electronic Railway Control and Protection Systems	
DIN	VDE 0831	Electrical Equipment for Railway Signalling	Railroad
	V VDE 0801	Principles for Computers in Safety-Related Systems	General
	V 19250	Fundamental Safety Analyses for MSR (Measurement-Control-Regulation) Protective Devices	

TABLE 3-1. MAJOR SAFETY VERIFICATION/VALIDATION STANDARDS/GUIDELINES FOR COMPUTER-BASED SYSTEMS (cont.)

Issuing Organization	Document Number	Title	Intended Application
Italian Railways	IS 402	Technical Specification for the Supply of Electronic Equipment for Safety and Signalling Systems	Railroad
BNCF/ AFNOR (Railway Standards Bureau)	NF F 71-011	Software Dependability-General Information	Railroad
	NF F 71-012	Software Dependability-Stresses on Software	
	NF F 71-013	Software Dependability-Adapted Methods for Software Safety Analysis	
Ministry of Defence	00-55 (Part 1)/ Issue 1	The Procurement of Safety Critical Software in Defence Equipment, Part 1: Requirements	Military
	00-55 (Part 2)/ Issue 1	The Procurement of Safety Critical Software in Defence Equipment, Part 2: Guidance	
	00-56/Issue 1	Hazard Analysis and Safety Classification of the Computer and Programmable Electronic System Elements of Defence Equipment	
IRSE	Report No. 1	Safety System Validation with Regard to Cross Acceptance of Signalling Systems by the Railways	Railroad
IEC	65A (Secretariat) 122 (Draft)	Software for Computers in the Application of Industrial Safety-Related Systems	Industrial
	65A (Secretariat) 123 (Draft)	Generic Aspects: Functional Safety of Electrical/ Electronic/Programmable Electrical Safety-Related Systems, Part 1: General Requirements	
	Std Pub 880	Software for Computers in the Safety Systems of Nuclear Power Stations	Nuclear
	987	Programmed Digital Computers Important to Safety for Nuclear Power Stations	
	62 (Secretariat) 69 (Draft)	Electrical Equipment in Medical Practice	Medical

TABLE 3-1. MAJOR SAFETY VERIFICATION/VALIDATION STANDARDS/GUIDELINES FOR COMPUTER-BASED SYSTEMS (cont.)

Issuing Organization	Document Number	Title	Intended Application
ANSI/IEEE	ANSI/IEEE-ANS-7.4.3.2-1982	Application Criteria for Programmable Digital Computer Systems in Safety Systems of Nuclear Power Generating Stations	Nuclear
	P-7.4.3.2, Draft 7	Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations	
	ANSI/ANS-10.4-1987	Guidelines for the Verification and Validation of Scientific and Engineering Computer Programs for the Nuclear Industry	
	ANSI/IEEE Std 1012-1986	Standard for Software Verification and Validation Plans	General
	IEEE Std. 603-1980	Standard Criteria for Safety Systems for Nuclear Power Generating Stations	Nuclear
	P 1228, Draft 5	Standard for Software Safety Plans	General
DOD	MIL-STD-882 B/C	System Safety Program Requirements	Military
	DOD-STD-2167A	Defense System Safety Development	
	MIL-STD-SDD (Draft)	(Revision to DOD-STD-2167A)	
	DOD-STD-2168	Defense System Software Quality Program	
US Air Force	AF REG 122-9	The Nuclear Safety Design Certification Program for Nuclear Weapon System Software and Firmware	Nuclear Weapons
	AF REG 122-10	Safety Design and Evaluation Criteria for Nuclear Weapon Systems	
	AFSC/AFLS Pamphlet 800-5	Software Independent Verification and Validation (IV&V)	Military
FAA	AC25.1309-1A	System Design and Analysis	Aviation
	RTCA/DO 178B	Software Considerations in Airborne Systems and Equipment Certification	

TABLE 3-1. MAJOR SAFETY VERIFICATION/VALIDATION STANDARDS/GUIDELINES FOR COMPUTER-BASED SYSTEMS (cont.)

Issuing Organization	Document Number	Title	Intended Application
SAE	ARP 4754, Draft 23C	System Integration Requirements	General/ Aviation
	ARP 4761, Draft 4	Safety Assessment Guidelines for Civil Airborne Systems and Equipment	
NASA	- - - (Draft)	Software Safety Standard	Aerospace
	SSP 30309 (Rev. B)	Safety Analysis and Risk Assessment Requirements Document	Space Station
	- - -	(Draft) Payload Requirements Document	Space Shuttle
JPL	D-576	Independent Verification and Validation of Computer Software: Methodology	Aerospace
	D-10058	Software System Safety Handbook	
FDA	- - -	Reviewer Guidance for Computer Controlled Medical Devices Undergoing 510(K) Review	Medical
AECL/ CANDU/ Ontario Hydro	982 C-H69002-0001	Standard for Software Engineering of Safety Critical Software	Nuclear
NBS	FIPS PUB 101	Guideline for Lifecycle Validation, Verification and Testing of Computer Software	General
	FIPS PUB 132	Guideline for Software Verification and Validation	
EIA	SEB6-A	System Safety Engineering in Software Development	General
NATO	STANAG 4404 (Draft)	Safety Design Requirements and Guidelines for Munition-Related Safety Critical Computing Systems	Munitions
	STANAG 4452 (Draft)	Safety Assessment of Munition-Related Computing Systems	

TABLE 3-1. MAJOR SAFETY VERIFICATION/VALIDATION STANDARDS/GUIDELINES FOR COMPUTER-BASED SYSTEMS (cont.)

Issuing Organization	Document Number	Title	Intended Application
European Space Agency	ESA PSS-05-01 Issue 2	ESA Software Engineering Standards, Issue 2	Space
	ESA PSS-01-40 Issue 2	System Safety Requirements for ESA Space Systems and Associated Equipment	
BSI	BS89/33005DC	Functional Safety of Programmable Electronic Systems: Generic Aspects	General
	BS89/33006DC	Software for Computers in the Application of Industrial Safety-Related Systems	
	B5587	Code of Practice for Testing of Computer-based Systems	
Underwriters Laboratory	UL1998	A Standard for Safety-Related Software	Consumer Products

Functional (Detailed Design) Specifications are written for each subsystem or function of the product. These specifications are then analyzed for hazards by Design Engineering.

Boards are laid out with circuits designed to meet the requirements of the Functional Specifications. Subsequently, failure modes and effects analyses (FMEAs) are performed on all boards, and designs are revised to eliminate any hazards/problems identified.

3.1.1.2 Software Design - GRS uses two special design methods to ensure the safety of software: Safety Assurance Logic (SAL) and Numerically Integrated Safety Assurance Logic (NISAL). One or both of these methods is used in all of their safety critical computer-based products. SAL entails the use of diversity, cycle checking and checkwords which can be generated only by the proper completion of certain critical tasks. With SAL, all data used by the system is represented by unique 32-bit values known as checkwords. Each operation performed on one or more pieces of data generates a new checkword. The failure to perform any operation, or the performance of any operation in the wrong sequence, will create an invalid checkword. NISAL is a variation of SAL in which the checkwords indicate both the identity and value of a parameter.

A separate computer, the Vital Power Controller (VPC) for Micro-Cabmatic products, or the Vital Relay Driver (VRD) for the Vital Processor Interlocking (VPI), will remove power from all vital outputs if an unsafe system failure is detected. The processor receives a "System Status" checkword from the main system processor once during each system cycle. This checkword is created by various vital functions such as the clearing of certain memory locations and verifying that no outputs have been incorrectly turned on. The system status checkword allows the VRD or VPC to produce vital output power for one system cycle. The absence of this checkword will remove power from all vital outputs.

3.1.1.3 Proof of Safety Documentation - The safety of the system software is documented in the "Proof of Safety" (POS) created for that product. There are two key parts to the POS: a General Part and a Specific Part. The General Part, consisting of Chapters 1 through 4, is similar for all products. The Specific Part, Chapter 5, is unique for each product.

Chapter 1 contains a complete list of all known failure types which can compromise the safety of the system. Examples include hardware failure, incorrect data (e.g. incorrect data stored in memory, data corrupted by noise, old data, or hardware failure of memory) and failure in data processing.

Chapter 2 contains a set of guidelines to be followed by the designer. It points out any factors which should be considered in the design.

Chapter 3 presents a menu of techniques for SAL which can be used to ensure safety in case of failures. Each technique also includes a discussion of the effectiveness of that technique to specific failures. The designer can pick one or more of the techniques to provide assurance of safety for each failure mode of the system.

Chapter 4 presents a menu of techniques for NISAL and includes a discussion of the effectiveness of the techniques. The designer can pick one or more of these techniques to provide assurance of safety for each failure mode.

Chapter 5 contains the failure analyses of the hardware and software for the specific product or system. The designer identifies how the design protects against each failure type listed in Chapter 1, identifying which technique from Chapter 3 (or Chapter 4) was used, including details showing that the technique was properly applied. The techniques chosen to eliminate hazards created by specific failures for a given product depend in part on the use of the product, and on its hardware design. Specific techniques used are a function of the product design, and not of the process used in mitigating hazards. The software design and the POS are reviewed by the designer's peers to ensure that all possible failures have been considered.

3.1.1.4 Application Design - Special design/verification activities are performed relative to application-specific design for different computer-based products. Two examples are given here -- one for the computer-based interlocking VPI and the other for the computer-based Micro-Cabmatic cab signalling system.

3.1.1.4.1 VPI Application Design Verification - GRS Application Engineering (AE) Department has a standards manual for procedures to be used in the application of VPI. A checklist must be filled out confirming that specific checks have been performed. After the designer has finished writing the Boolean expressions defining the logic for the location, a PROM code is generated. The input file created by the designer is compiled by the VPI CAA program to produce the PROM files. In order to ensure that the PROM file was properly generated, an independent program known as the Application Data Verifier (ADV) decompiles the PROM file and generates a new listing of Boolean expressions. The circuit (logic) check is then performed on the output listing from the ADV. The ADV program was written using completely different code from that in the CAA package. This ensures that any compilation errors made by the CAA cannot be corrected by the ADV. These software verification activities are performed by an AE Design Verification Group.

Application wiring is checked by individuals other than those who designed the circuits.

Certain information, in addition to the application logic, is placed in the PROM file by the CAA package to ensure data integrity. In addition, the compiler performs a Cyclic Redundancy Check (CRC) on the PROM file and stores the result in the file. This enables the VPI software to continuously check the integrity of the application program while the system is operating. The CRC check of program code is compared with the CRC check made by the compiler. Any errors detected will cause the system to shut off all vital outputs.

3.1.1.4.2 Micro-Cabmatic Application Design Verification - Application software for Micro-Cabmatic is designed based on written specifications from the customer. Data structures are required for items such as cab signal code rates, actual speed limits and gear-box tooth spacing. Application software is verified primarily by rigorous testing in the lab. All code rates and speeds are simulated and system response is monitored. An ADV is currently being developed for Micro-Cabmatic.

3.1.1.5 System Testing - Both factory and field tests are performed on the system. Breakdown tests are performed on all wiring. Logic is tested by manipulating inputs and verifying the states of the outputs. Testing is performed not only to confirm that the system will perform in the desired manner, but also to verify that it will not permit unsafe conditions. For example, in the case of VPI, testing would be conducted to verify that until time locking had expired for one route, a conflicting route could not be established.

3.1.1.6 Reverification of Modifications - GRS has specific procedures for replacing/upgrading PROMs in the field. For computer-based interlockings, these procedures require the recording, in a configuration log, of certain data that can be read by means of a hand-held terminal when the equipment is placed into service. Then, when a PROM is replaced, this data must again be read and compared with that in the log. Included in the data are checksums which verify the correctness of the system and application software programs. This data must be identical for the new PROM to be placed into service. When software is changed, the revision signature must be changed, and jumpers on the motherboard must be changed to reflect the new revision. The system will operate only if the jumper setting agrees with the signature programmed into the software. Also, the PROM code from the revised software is run through the ADV discussed earlier. The output from the ADV is compared to that from the earlier version of the program to ensure that no part of the program was inadvertently changed.

When input and output boards are replaced, proper operation is reverified. Inputs are cycled on and off, and the TRUE/FALSE states of the variables are observed using the hand-held terminal. Likewise, outputs are controlled ON and OFF, and the operation of the outputs is observed.

Testing of system boards (e.g., CPU board) is not required after replacement, other than the verification of the software as described above. The system software ensures the vitality of this hardware.

3.2 UNION SWITCH AND SIGNAL

Union Switch & Signal (US&S) has headquarters in Pittsburgh, Pennsylvania. The design, verification and validation of its vital products and systems are generally carried out by the Research & Development Department. The Product & Technology Assurance (P&TA) Department plays an oversight role in ensuring compliance with the safety design and implementation requirements of these systems.

US&S's product/system safety design methodology (including safety verification and validation), for vital products and systems is based on the use of specific design guidelines as well as analyses and tests which are conducted throughout the life cycle of the product/ system. General design guidelines are followed in the design of all vital and non-vital equipment/systems, and more specific mandatory guidelines are followed for vital equipment and systems. These guidelines take into account established industry practices and standards such as those promulgated through the FRA, AAR, IEEE, ASME, NEMA, ANSI, NTSB and other appropriate bodies.

In the case of computer-based equipment, even more specific guidelines are followed. This equipment is designed with traditional vital (Class I) hardware as well as non-vital (Class II) hardware controlled by software. The operation of the hardware and software elements is continuously checked by extensive, layered diagnostics utilizing diversity and self-checking principles. Some key general assumptions used by US&S in the design are as follows:

- The Class I hardware portions of the equipment consist of traditional analog circuits with discrete components that have well-understood fault modes. The design of these circuits is such that every credible single-point fault mode is self-revealing and results in a safe-side behavior at the equipment level.

- The Class II hardware, consisting of integrated circuits of various complexities and usually with unknown or not-so-well understood fault modes at the physical device level, will exhibit a finite number of fault modes at the gate level and at the external pin level. Fault modes at these levels can be counteracted with diversity and self-checking principles, so that single-point, common-cause, transient, permanent, near-coincident faults in hardware or software, due to internal or external causes, are effectively controlled from producing unsafe behavior at the equipment level.

- The subset of faults in any of the categories listed above that could defeat the extensive, layered diagnostics in such a way as to result in an undesirable outcome is so small ("improbable" category per MIL-STD-882B) that an acceptable lower bound on safety is ensured. As described below, a quantitative evaluation on the lower bound on safety can be made.

3.2.1 Safety Verification and Validation Process

The basic design methodology for the elimination and control of hazards is shown in Figure 3-1. This basic approach is used by US&S in the design, review and implementation of all vital products including those incorporating computers to perform vital functions. Safety verification and validation activities are an integral part of this overall approach. As can be observed from the figure, one of the early activities that is initiated is a Preliminary Hazard Analysis (PHA). The purpose of this analysis is to identify potential hazards that could be associated with all phases of the product life cycle. Other analyses that are typically performed include a Subsystem Hazard Analysis (SSHA), Hazard Modes and Effects Analysis (HMEA) or Fault Hazard Analysis (FHA), an Interface Hazard Analysis (IHA), and Operating and Support Hazard Analysis (O&SHA). Many of these analyses are similar to those discussed in MIL-STD-882B/C. In addition, engineering prototype testing of hardware and software is conducted in the laboratory under simulated field conditions. Based on the results of these design-related assessments, modifications are made or other measures are taken as necessary to eliminate risks or reduce them to acceptable levels. Changes to hardware and/or software are analyzed and tested as deemed necessary.

After the product or system is manufactured, a pilot or production unit is re-verified in the lab and/or field. System and acceptance testing is conducted after installation. Further, the safe operation/performance of the product/system is tracked during its service life.

In some instances, a quantitative evaluation of the lower bound on safety is made with the aid of formal methods and a computerized design, verification and validation toolset such as US&S's D RAMP Toolset D RAMP consists of analog and digital simulation software packages, Instruction Set Architecture models of microprocessors using VHDL, and Markov Process models of system behavior under fault conditions. This toolset is currently being formalized by US&S for application to all computer-based vital products.

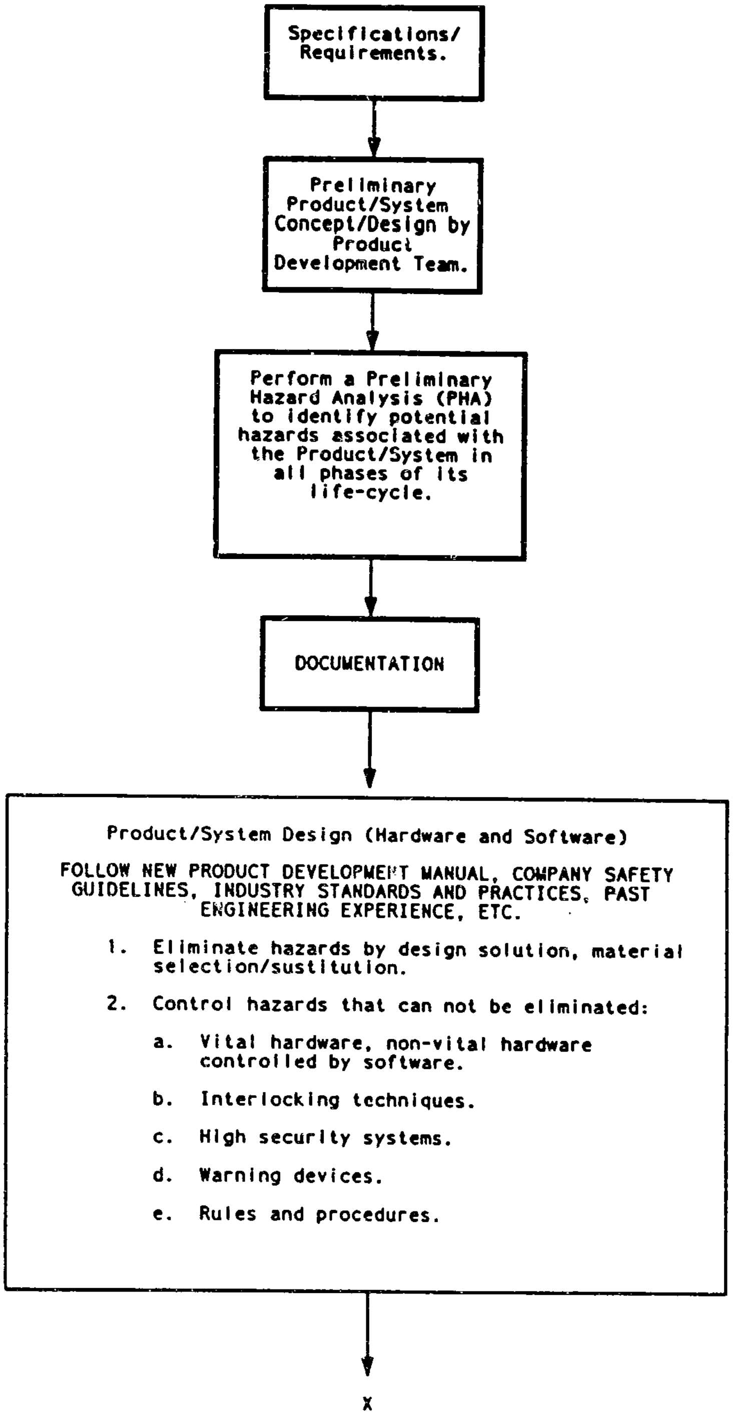

Specifications/ Requirements.
Preliminary Product/System Concept/Design by Product Development Team.
Perform a Preliminary Hazard Analysis (PHA) to identify potential hazards associated with the Product/System in all phases of its life-cycle.
DOCUMENTATION
Product/System Design (Hardware and Software)
FOLLOW NEW PRODUCT DEVELOPMENT MANUAL, COMPANY SAFETY GUIDELINES, INDUSTRY STANDARDS AND PRACTICES, PAST ENGINEERING EXPERIENCE, ETC.
1. Eliminate hazards by design solution, material selection/sustitution.
2. Control hazards that can not be eliminated:
a. Vital hardware, non-vital hardware controlled by software.
b. Interlocking techniques.
c. High security systems.
d. Warning devices.
e. Rules and procedures.
X

FIGURE 3-1. US&S PRODUCT/SYSTEM SAFETY DESIGN PHILOSOPHY

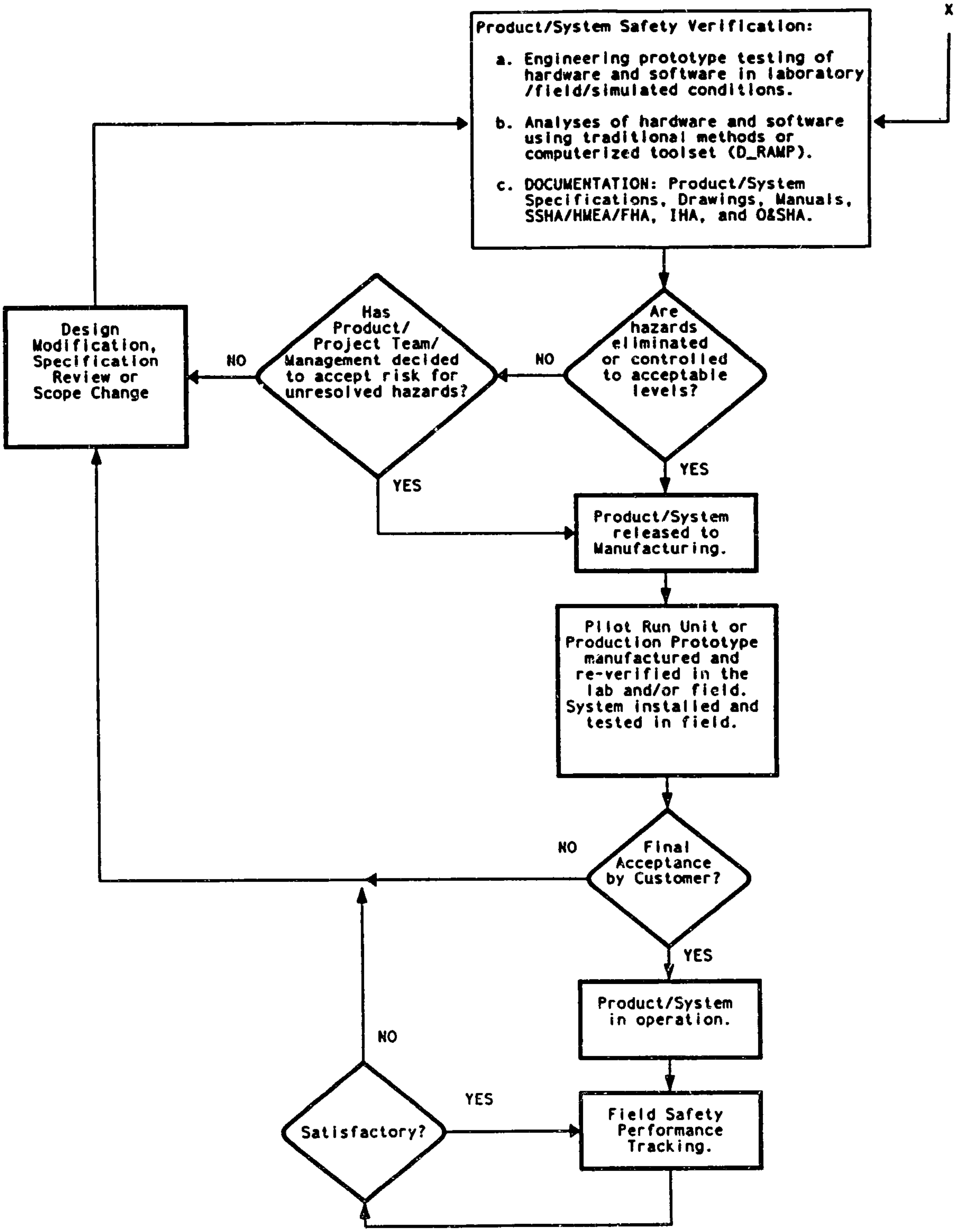

X
Product/System Safety Verification:
a. Engineering prototype testing of hardware and software in laboratory /field/simulated conditions.
b. Analyses of hardware and software using traditional methods or computerized toolset (D_RAMP).
c. DOCUMENTATION: Product/System Specifications, Drawings, Manuals, SSHA/HMEA/FHA, IHA, and O&SHA.
Design Modification, Specification Review or Scope Change
NO
Has Product/ Project Team/ Management decided to accept risk for unresolved hazards?
NO
Are hazards eliminated or controlled to acceptable levels?
YES
YES
Product/System released to Manufacturing.
Pilot Run Unit or Production Prototype manufactured and re-verified in the lab and/or field. System installed and tested in field.
NO
Final Acceptance by Customer?
YES
Product/System in operation.
NO
YES
Satisfactory?
Field Safety Performance Tracking.

FIGURE 3-1. US&S PRODUCT/SYSTEM SAFETY DESIGN PHILOSOPHY (cont.)

3.2.1.1 Modifications - Reverification of safety is recommended/conducted by US&S following post-development modifications. For example, in the case of computer-based interlockings, a full interlocking operational test is recommended following any changes to safety critical software. If hardware changes are made, it is recommended that the affected circuit boards are retested and/or reanalyzed based upon specific procedures developed by US&S.

3.3 HARMON ELECTRONICS

Harmon Electronics, Inc., with Headquarters in Grain Valley, Missouri, supplies vital (safety critical) and non-vital equipment/systems (including vital computer-based products) to the railroad and transit industries. As part of Harmon's effort to produce functionally correct and safe products that meet the needs of their customers, an internal Product Safety Program Plan has been developed. This plan outlines the safety-related activities which are performed throughout the product development life cycle from concept through post-development operation/maintenance. Roles/responsibilities and documentation requirements are also defined within the plan.

Figure 3-2 shows Harmon's overall engineering development cycle and associated major life cycle phases from concept (based upon customer requirements) through product release to manufacturing/production. The various activities conducted during this development process to ensure product safety, including safety verifications/validations of hardware and software, are summarized in Table 3-2 along with their purpose. Table 3-2 also describes safety-related activities performed after product release in the following phases: production/manufacturing, shipping, field service/customer support, and field service/product support.

Harmon also performs a number of process management related activities to continually improve their overall safety assurance process. This includes safety training of appropriate company personnel, the establishment of a Corporate Product Assurance Team (CPAT) to resolve relevant safety issues, the conduct of program reviews and audits to assure synchronization of product safety program activities with corporate goals and objectives, and others.

3.4 ALCATEL-CANADA

ALCATEL-Canada's SEL Division, located in Weston, Ontario, is a supplier of signalling and train control systems/equipment worldwide, including safety critical computer-based products. However, only very general information pertaining to ALCATEL's safety verification and validation process was available at the time of this report.

ALCATEL performs a number of activities in order to ensure the safety, reliability and maintainability of vital and non-vital products. This includes the conduct of FMEAs, Preliminary Hazard Analyses, Fault Tree Analyses and in-process audits. All quality assurance (QA) activities which include testing and V&V usually are initiated early in the product life cycle (i.e., concept or requirement phase) and continue throughout development.

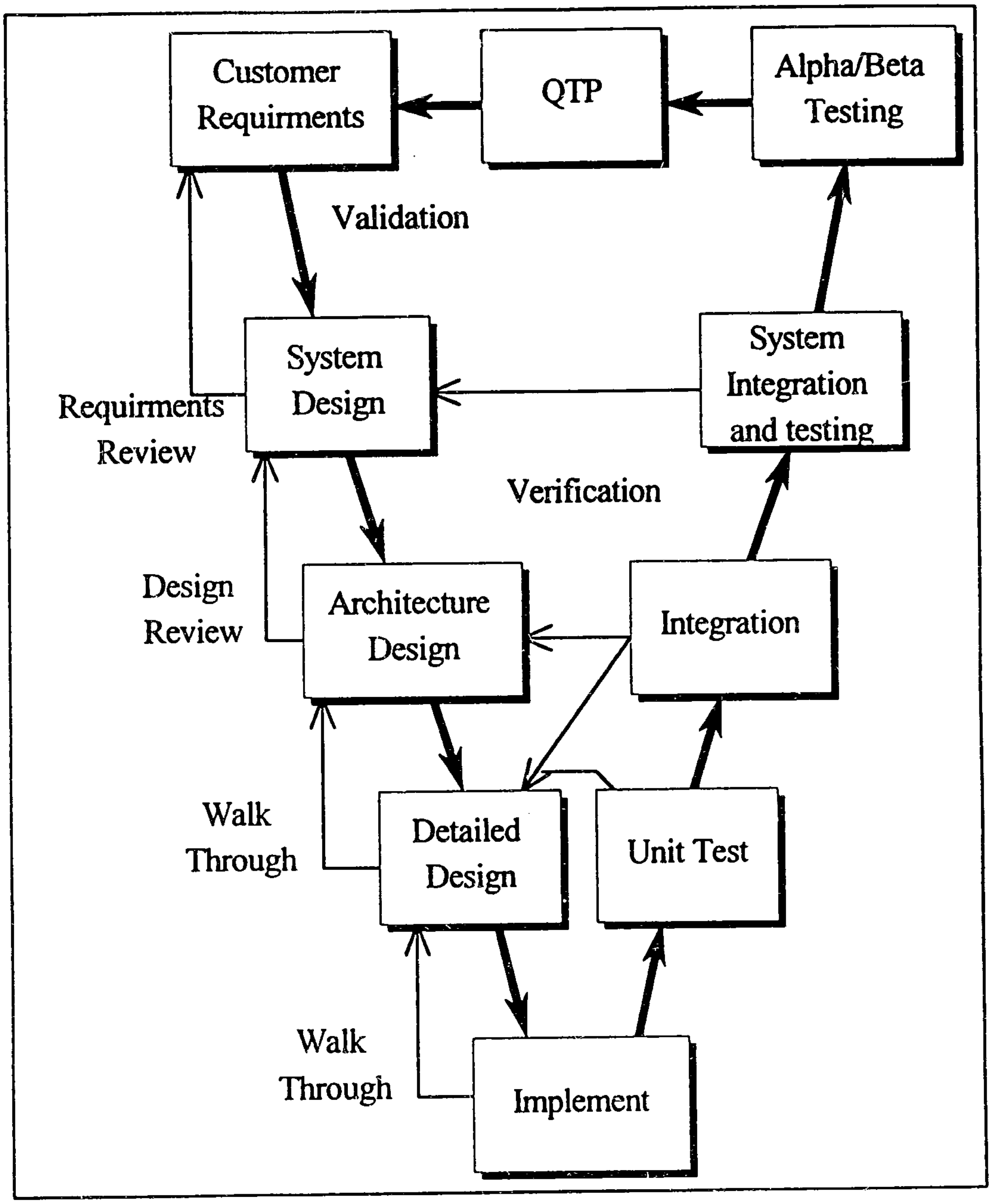


FIGURE 3-2. HARMON'S ENGINEERING AND DEVELOPMENT CYCLE

TABLE 3-2. HARMON'S MAJOR PRODUCT DEVELOPMENT LIFE CYCLE PHASES AND ACTIVITIES

Phase	Activity/ Documentation	Purpose
Customer Requirements	Product Requirements	Establish what product is to do.
	Contract Review (Requirements Review)	Allow input into the preliminary requirements from cross functional areas.
System Dcsign	System Specification	Working document that lists in plain English the requirements of the unit (both functional and safety).
	Safety Criteria	A living document that contains lessons learned, recommended practices and safety architecture specific criteria.
	Reverse Engineering	Done for existing products that apply to the current product under development. To ensure that any requirements that are in existing products will be evaluated and transferred to the new product if deemed necessary.
	Product Manuals, Dangers, Warnings, Cautions and Notes	All product, technical and other manuals and sales applications literature shall be reviewed for adequate inclusion of dangers, warnings, cautions, notes and other safety information as deemed appropriate.
	Preliminary Hazards Analysis (PHA)	Analysis that investigates, categorizes and classifies, inherent, functional and operational & maintenance hazards.
	System Test Plan	Establish a test plan to ensure that system meets the system requirements.
	System Design Review	Periodic review of the progressing design with members of various disciplines depending on the amount of detail. Safety is an agenda item at each review.
	Operations and Maintenance Analysis	Done as part of the PHA to uncover potential O&M problems.

TABLE 3-2. HARMON'S MAJOR PRODUCT DEVELOPMENT LIFE CYCLE PHASES AND ACTIVITIES (cont.)

Phase	Activity/ Documentation	Purpose
Architecture Design	Architecture Specification	Expanded working document that lists in plain English the requirements of the unit (both functional and safety). Sometimes referred to as an expanded system specification.
	Fault Tree Analysis (FTA)	Delineation of Class 1 & 2 functional hazards identified in the PHA. This output of the FTA provides safety criteria for individual software and hardware modules.
	Design Review	Periodic review of the progressing design with members of various disciplines depending on the amount of detail. Safety is an agenda item at each review.
Detailed Design	Hardware Detailed Design Specification	Expanded working document that lists hardware requirements of the unit (both functional and safety) and shows how these requirements will be implemented.
	Software Detailed Design Specification	Expanded working document that lists the software requirements of the unit (both functional and safety) and shows how these requirements will be implemented.
	Detailed Design Review	Periodic review of the progressing design with members of various disciplines depending on the amount of detail. Safety is an agenda item at each review.
Implementation	Failure Mode and Effects Analysis (FMEA)	Detailed analysis of hardware failure modes and their effect on the system. Performed for all hardware elements determined by FTA.
	Software Failure Mode and Effects Analysis (SFMEA)	Detailed analysis of software failure modes and their effect on the system. Performed for all software elements determined by FTA.
	Code Walkthrough	Periodic review of the progressing design with members of various disciplines depending on the amount of detail. Safety is an agenda item at each review.
	P.C. Board Trace Analysis	Ensures vital separation of those trace elements determined to be vital. Elements are determined by FTA, FMEA or design review.
Unit Test	Code Test Record	Tests that are performed to ensure proper software function for a specific routine. 100% test coverage is the goal at this level.
	P.C. Board Test Procedure	Test procedure used by production to ensure proper functionality of each hardware module.

TABLE 3-2. HARMON'S MAJOR PRODUCT DEVELOPMENT LIFE CYCLE PHASES AND ACTIVITIES (cont.)

Phase	Activity/ Documentation	Purpose
Integration	Integration Tests	Tests that are performed during integration to reveal any hidden (latent) interface faults.
System Integration and Testing	System Test Plan	Tests that are performed to ensure that the systems meet the initial customer requirements.
	FMEA Testing	Physical shorts and opens test of board to ensure that it meets its safety requirements.
	SFMEA Testing	Testing of the software by inserting failure modes (both hardware and firmware generated) into the unit to ensure the software behaves as predicted.
Alpha/Beta Testing	Problem Reports	Database of reported errata from the field or lab. Identifying who reported the problem and the conditions it was reported under.
	Corrective Action and Control	Implantation of fixes or work-arounds for reported problems. Then, incorporating final fixes during subsequent releases.
Release	Qualification Test Procedures (QTP)	Excerpts from all testing phases that are completed for each release. The QTP covers both safety and functional requirements.
	Document Control and Configuration Management	Configuration management for all products will be maintained through a corresponding Engineering Change (ECR/ECO) system. Controls will be implemented and maintained so that the configuration of each product is known and documented at all times. No changes will be allowed to any design, parts, lists of materials, or any other feature which could affect product form, fit, or function, without approval of the responsible product design manager through an ECR/ECO. Any time a change to form, fit or function is proposed, a complete and thorough safety analysis of the proposed change will be made and documented.

TABLE 3-2. HARMON'S MAJOR PRODUCT DEVELOPMENT LIFE CYCLE PHASES AND ACTIVITIES (cont.)

Phase	Activity/ Documentation	Purpose
Production/ Manufacturing	Vendor Qualification	Certain elements of the safety plan may be allocated to individual subcontractors and the equipment/subsystems provided by them. Imposition of these elements will be by contract, with the specific requirements included in the contract procurement specifications.
	Quality Assurance (QA)	First level control will be through quality verification tests and audits conducted by the individual subsidiary Quality Assurance process at various stages throughout the product manufacturing process. An additional control point will be a final audit of each end item by the Quality Assurance Department before shipment, in accordance with specified product test plans and audit criteria. A third level of control will be a sampling audit of finished products on a random basis.
	Test Records	These are tests that are required by engineering on end item production units to ensure that no latent manufacturing faults exist.
Shipping	Dangers, Warnings, Cautions and Notes	Certain inherent hazards exist in packaging, shipping and storage of equipment. Shipping cartons will be reviewed for proper inclusion of the appropriate warning labels.
	Sales Literature. Dangers, Warnings Cautions and Notes	All product, technical and other manuals and sales applications literature shall be reviewed for adequate inclusion of dangers, warnings, cautions, notes and other safety information as deemed appropriate.
Field Service/ Customer Support	Technical Manuals, Installation Procedures and Drawings	All product, technical and other manuals and sales applications literature shall be reviewed for adequate inclusion of dangers, warnings, cautions, notes and other safety information as deemed appropriate.

TABLE 3-2. HARMON'S MAJOR PRODUCT DEVELOPMENT LIFE CYCLE PHASES AND ACTIVITIES (cont.)

Phase	Activity/ Documentation	Purpose
Field Service/ Product Support	Problem Reports	Database of reported errata from the field. Identifying who reported the problems and the conditions it was reported under.
	Maintenance	All product, technical and other manuals and sales applications literature shall be reviewed for adequate inclusion of dangers, warnings, cautions, notes and other safety information as deemed appropriate.
	Corrective Action and Control (Accident Investigation, Incident Resolution)	The Harmon Industries Incident Resolution Working Team will support the Corporate Product Assurance team by responding to incidents, assessing risk and assuring corrective action is taken in a timely manner which meets the safety and reliability objectives of both our internal and external customers.
	EIB/ESB	Information regarding product improvements, safety notifications, etc. will be transferred to the customer via EIB's/ESB's. EIB's are typically informational only. ESB's are used for safety and reliability.
	Configuration Management	Configuration management for all Harmon Industries, Inc. products will be maintained through a corresponding Engineering Change (ECR/ECO) system. Controls will be implemented and maintained so that the configuration of each product is known and documented at all times. No changes will be allowed to any design, parts, lists of materials, or any other feature which could affect product form, fit, or function, without approval of the responsible product design manager through an ECR/ECO. Any time a change to form, fit or function is proposed, a complete and thorough safety analysis of the proposed change will be made and documented.

Quality assurance tasks are performed by two dedicated groups: Software Quality Assurance and System Assurance. Further, the Engineering team has an independent V&V group and System Reliability group. There are also dedicated groups for quality control, test and commissioning of all products.

3.5 ADVANCED TRAIN CONTROL SYSTEM

The Advanced Train Control System (ATCS) is a computer-based command, control and communications system that is being developed as a joint venture of the Association of American Railroads (AAR) and the Railway Association of Canada (RAC). It is based upon the use of wayside transponders and on-board odometers for train location/speed and a 900 Mhz radio data communications link. Computers will be used extensively in the implementation of both safety and non-safety critical functions.

At the time of this report, a set of 31 specification documents have been generated through a public open-forum process involving contracted systems engineers, suppliers and other railroad industry personnel. The purpose of these specifications is to provide standardization of the performance and interface requirements of ATCS hardware and software without limiting internal design approaches of suppliers.

There are two main specifications which are particularly relevant to the topic of this study. One pertains to system safety assurance and the other to software quality assurance. They are:

- "Recommended Practices for Safety and Systems Assurance," ATCS Specification 140, Revision 3.0, March 1993, and

- "Recommended Practices for Software Quality Assurance," ATCS Specification 130, Revision 3.0, March 1993.

The former defines the minimum requirements for the nature and content of a Safety and Systems Assurance Program Plan, while the latter describes the requirements and process for achieving software quality assurance. Brief overviews of these two specifications are provided below, with emphasis on Specification 140 (since it pertains directly to safety).

The original concept around which the above specifications were written is that each railroad and/or system integrator would be responsible for the safety of the equipment used on that railroad and each supplier would develop and demonstrate the safety of their own system including the software. Within the last year, consideration is being given (by the AAR and others) to changing this concept involving the software to one in which there would be an industry funded "single set" of software for safety-related applications within the ATCS. In this new concept, one contractor would be chosen to develop the software according to ATCS specifications, and a second contractor would be responsible for the independent verification and validation (IV&V) of that software. Thc IV&V effort would include the following four activities:

- Formal review of Control Flow Specifications and other software documentation

- Formal review of all source code

- Formal review of all test results

- Safety audit of software design and implementation including the checked redundancy mechanism.

Completed software and documentation would be delivered to the AAR, who would then provide copies of the software to component manufacturers and/or systems integrators. At the time of this report, this new concept was not finalized — it was only under consideration. Acceptance of this concept may require some modification of the two specifications identified above (i.e., 130 and 140) as well as others.

One recent activity pertaining to this new concept was the release by the AAR of a Request for Information (RFI) entitled "ATCS Industry Standard Software Development." The purpose of this RFI was to solicit information for a cost/benefit analysis that would assist in this overall decision making process regarding the new software development concept.

The Canadian Government has also conducted some studies of safety issues involving the use of computers in ATCS systems. In November 1990, Transport Canada (of the Railway Safety Directorate) issued the document TP 10770E, "ATCS System Safety Validation Programs." That report is the result of a study conducted by Queen's University for the Transportation Development Center (TDC), the research arm of Transport Canada. Its primary purpose was to investigate the need for, and to identify and evaluate, possible ATCS system safety validation programs. Objectives of such a validation program were as follows:

- Demonstrate that equipment conforms to specifications

- Demonstrate that equipment of different suppliers and railways is operable over dark and signalled territory

- Demonstrate that equipped and non-equipped systems are operable over dark and signalled territory

- Verify and validate specifications.

The results of the study was the recommendation that an ATCS system safety validation program should be established based upon a System Integration Simulator/Emulator/Tester (SISET) approach. This would permit the simulation/emulation of ATCS system elements in order to perform static and dynamic control flow validation and system interoperability tests, and to test equipment to ensure it meets specifications and is compatible with other vendor's equipment.

3.5.1 ATCS Specification 140

This specification, "ATCS Recommended Practices for Safety and Systems Assurance," defines the nature and content of a Systems and Safety Assurance Program Plan (SSAPP) for ATCS applications. The SSAPP is to define the essential requirements for the design, development and implementation of ATCS components and systems relative to five major areas: safety, reliability, maintainability, quality assurance and human factors. Separate program plans are to be established for each of these five areas, and are to be included within the overall SSAPP.

In the area of safety, a System Safety Program Plan (SSPP) is to be developed in order to help ensure that all reasonable steps are taken to minimize the risk of loss, damage or injury to personnel or property. The major portion of the System Safety Program Plan, Section (2.0) of Specification 140, discusses safety analysis/testing tools and techniques which could be applied (by a supplier) throughout an ATCS program life cycle to help ensure safety. These tools/techniques are presented as examples, and are based upon those described in MIL-STD-882B. The supplier is encouraged to select the appropriate techniques according to the specific system component and the railroad safety requirements, and to refer to 882B for further guidance. The tools/techniques are divided into two main sections: Design and Evaluation Tasks, and Verification and Testing.

3.5.1.1 Design and Evaluation Tasks - A number of activities/techniques are discussed that could be used concurrently with the equipment development process. Those are briefly discussed below:

- Preliminary Hazard List -- a list of possible hazards, generated early in the design

- Preliminary Hazard Analysis -- conducted early to establish an initial risk assessment

- Subsystem Hazard Analysis -- conducted as soon as possible on subsystems to identify associated hazards including component failure modes, human error inputs and others

- System Hazard Analysis -- determines safety problem areas of total system design

- Fault Effects Analysis (FEA) -- reviews system outputs to determine potentially hazardous effects; a fault tree technique is suggested

- Failure Modes, Effects and Criticality Analysis (FMECA) -- considers failure modes, their effects, criticality of the failure (impact on safety, mission success and demand for maintenance support) and failure indications/annunciations

- Software Requirements Hazard Analysis (SRHA) – thorough review and analysis of software safety requirements to ensure proper translation into software specifications

- Top-Level Design Hazard Analysis (TDHA) – includes relating earlier identified hazards to software components, identifying safety critical computer software components (SCCSCs), examining software for independence/dependence and interdependence among software components, analyzing software elements that can influence SCCSCs for undesired effects, and analyzing top-level design of SCCSCs for compliance with safety requirements

- Detailed Design Hazard Analysis (DDHA) – includes relating hazards to low level software components, analyzing software components that can affect or influence SCCSCs for correctness and undesired effects, analyzing detailed design of SCCSCs for compliance with safety requirements, developing requirements for inclusion in test plans and other areas (e.g., system documentation), and providing coding personnel with recommendations

- Code Level Software Hazard Analysis (CSHA) – examines actual source and object code to verify design implementation; starts when coding commences and is continually updated; identifies actions required to eliminate identified hazards or reduce risk; looks at SCCSCs for correctness and sensitivities (e.g., timing, I/O problems); looks at programs, routines, modules, etc. for design/coding errors; examines software of SCCSCs at all levels for compliance with safety criteria; examines single point and likely multiple failure effects on inputs; examines combinations of hardware and software failures, unintended program jumps, etc; examines out-of-bounds or overloading input conditions; includes review of software documentation.

3.5.1.2 Verification and Testing - A program of testing/demonstrations is to be defined in order to verify compliance of safety critical hardware, software and procedures with safety requirements. Testing is also needed in instances where analyses or inspections cannot show that risk is acceptably reduced.

3.5.1.2.1 Safety Evaluation and Test - This task is to conduct safety verification testing as part of component engineering development, during prototype evaluation and during ATCS pilot test bed programs. It is to be a part of the design, production, and operation and maintenance phases. In component development, testing should verify results of safety analyses conducted during the design phase – all potential hazards defined in the system hazard analysis should be addressed. Potentially unsafe combinations of input, signal and operational conditions (as identified by analysis) should be simulated. Multiple and dependent failure conditions should also be tested.

ATCS hardware and software prototypes should be tested in both laboratory and field environments. The purpose is to show that hardware and software will fail in a safe state when utilized in their operational environment.

3.5.1.2.2 Software Safety Testing - Another task is to focus on testing the lower level units of the software. Objectives here include the following:

- Ensure identified safety hazards have been eliminated or their risk reduced
- Ensure SCCSCs are tested properly and results are documented
- Test software under abnormal environmental and input conditions as well as normal conditions
- Ensure via stress and acceptance testing that software operates safely under worst case conditions
- Ensure software not specifically developed for ATCS performs safely
- Ensure safety hazards and other concerns identified during system integration and acceptance testing are corrected and retested as necessary.

3.5.2 ATCS Specification 130

This document, "Recommended Practices for Software Quality Assurance," describes the requirements and overall process for achieving software quality assurance in an ATCS environment. In addition to describing an overall process for addressing quality ass. ance throughout the life cycle of software, it addresses related matters such as roles/ responsibilities of vendors/railroads, use of existing standards, milestones/reviews and configuration management.

Six major goals of software quality assurance are listed, with one being to help ensure safety of operations influenced by or controlled by ATCS computer-based equipment. As such, this document is intended to be used in conjunction with Specification 140, which (as described earlier) outlines the means of ensuring overall system safety, including that of the software. The overall objective of the software quality assurance process, according to this document, is to support the detection and correction of software defects throughout the developmental and operational life of the system.

Section 5 provides a description of the basic software quality assurance process. One key aspect of this process is that the Department of Defense standard, DOD-STD-2167A (Defense System Software Development), is recommended as a model for the development of the software. However, the process described in Specification 130 is actually a tailored version of that contained in DOD-STD-2167A. Activities, products/documentation and milestones are suggested for different software life cycle phases.

Following a discussion on configuration management, there is a description of the role of testing in the software quality assurance process. Testing is described as being the primary method of identifying and reducing errors in the software. It is suggested that the quality assurance process must rely on layers of structured, redundant and independent testing methods to obtain the acceptable level of confidence in software quality. Five levels of testing are described ranging from program/module tests to system performance tests.

There is also a discussion of the software acceptance process. For vital software, the process is to include testing by the developing organization, independent verification and validation by a separate body, and certification (establishing the functional completeness and performance of the software).

3.6 BRITISH RAIL

British Rail Research (BRR), located in Derby, England, has had the responsibility of underwriting safety critical signalling systems and approving them for operational use within the British Railways. Specifically, this responsibility has been in the Safety Critical Systems Unit of the Engineering Research and Development Department. This unit conducts safety verifications and validations of computer and non-computer-based equipment (developed by BRR), including validations of software based on International Electrotechnical Commission (IEC) standards/principles. In addition, they have been utilizing formal methods to prove safety. This same group within British Rail had primary responsibility for developing the Solid State Interlocking (SSI) system, a computer-based interlocking utilizing a triple redundant computer architecture with majority voting. Other key organizations in that development included GEC General Signal and Westinghouse Signals.

The Railway Inspectorate (equivalent to the FRA in the U.S.) is part of the Health and Safety Executive (H&SE), and is playing a major role in the oversight and regulation of safety in England. Manufacturers and users (e.g., British Rail) must produce a proof-of-safety or "safety case" which provides the justification for the safety of a system. This safety case is to cover the entire life cycle of the system from concept through maintenance. The Railway Inspectorate will audit the safety evidence provided and expect the manufacturer/BRR to follow the best available standards.

The H&SE and the Interdepartmental Committee on Software Engineering (ICSE) have issued several documents including the following which, although advisory/informative in nature and not mandatory standards, do address the safety of computer-based systems:

- "Programmable Electronic Systems in Safety-Related Applications, Part 1-An Introductory Guide" and "Part 2-General Technical Guidelines," H&SE. (1987), and

- "SafeIT: A Government Consultation Document on the Safety of Computer Controlled Systems. Part 1-Overall Approach" and "Part 2-Standards Framework."

These documents were created with the intent of raising the awareness of the need to consider safety in computer-based systems.

The IEC, via two working groups (i.e. WG9 and WG10), has issued the following two draft standards which pertain to programmable electronic systems in general (and not uniquely to railway applications):

- 65A (Secretariat) 122, "Software for Computers in the Application of Industrial Safety Related Systems," (WG9), and

- 65A (Secretariat) 123, "Functional Safety of Programmable Electronic Systems: Generic Aspects," (WG10).

The draft 65A (Sec.) 122 describes procedures and possible techniques for developing and maintaining software in programmable electronic systems. It also describes associated documentation. Verification and validation activities are treated as an integral part of the development process. A key aspect of this document is the concept of four different integrity levels of safety-related software. Various V&V techniques are associated with each of these levels. It is intended for this draft to be used in conjunction with draft 65A (Sec.) 123. This latter document is more systems-oriented, and describes a general approach for all safety related activities to be performed during the life cycle of a system. The IEC documents are discussed in more detail in the IEC section of this report.

The British Standards Institution (BSI) has also issued two standards which mirror the existing IEC drafts listed above. In particular, the BSI standards BS89/33006DC and BS89/33005DC relate to 65A (Sec.) 122 and 65A (Sec.) 123, respectively.

The Railway Industry Association has issued a "Consultive Document" entitled "Safety Related Software for Railway Signalling," BRB/LU LTD/RIA Technical Specification No. 23: 1991. The intent of this document is to apply the procedures in IEC 65A, and especially those in (Sec.) 122 pertaining to software, to railway signalling. It is this document which is considered as a standard by the BRR. A brief overview of RIA 23 is provided later in this section, following an overview of BRR's safety V&V process.

3.6.1 Safety Verification and Validation Process

As described earlier, BRR performs safety verification and validation activities on equipment developed internally.

3.6.1.1 Software V&V - The V&V process, particularly for software, is based heavily upon the RIA 23 document described above. BRR has developed the following three major internal documents that describe their V&V policy and practices with respect to software:

- "Software Verification and Validation," SSU-D-SVA-RR-1, 27 November 1992
- "Code of Practice for Validation of Safety Critical Software," ELS-DOC-4888, Issue A, 26 October 1990, and
- "Code of Practice for the Validation of Modifications to Previously Validated Code," ELS-DOC-4817, Issue A, 6 September 1990.

Due to the proprietary nature of these documents, only brief overviews of their intent are provided here.

3.6.1.1.1 ELS-DOC-4888 - This document addresses the more general question of validating safety critical software post-design, and describes the techniques and procedures that are used by BRR. It includes a list of the documentation required as input to the process, instructions for performing the various forms of analysis (including structural, control flow, data and information flow, semantic and timing), and instructions for the production of the validation report and supporting documentation. The use of the SPADE validation tools is addressed. An appendix defines the design practices necessary to permit the analysis using SPADE tools of software written in assembler.

3.6.1.1.2 SSU-D-SVA-RR-1 - This document reviews the present policy of BRR for the V&V of safety critical software. It includes a systematic comparison between current BRR practice and appropriate industry, national and international standards, and identifies the actions needed to ensure full conformance with currently accepted "best practice."

3.6.1.1.3 ELS-DOC-4817 - This document defines the aims of validation in the context of changes to previously validated code, and formalizes the approach to be used within BRR to validate such changes. The procedures are intended to ensure that any changes made to existing code correctly implement the defined requirements without corrupting the functionality of those parts of the code which are not changed. The document contains instructions for performing structural, semantic and information flow analysis with these objectives in mind and describes how the results of these analyses should be used to formulate the final validation documentation. The procedures are directed toward the re-validation of software written in assembler.

3.6.1.1.4 Formal Methods - BRR also uses formal methods to demonstrate the safety of the software in certain specialized areas. Since signalling systems are data-driven, computer tools have been developed to help generate data and check the consistency of the data. Attention in this area is being directed to more automated checking techniques.

3.6.1.2 Hardware/System Validation - BRR's specific hardware/system level V&V documentation is also proprietary. However, demonstrating redundancy management (i.e., how the safety protection mechanisms work) is of utmost importance. Fault tree analyses and FMECAs are initiated early to assist in making design decisions, and then updated later to assess the safety of the design. The interaction of software and hardware also receives significant attention via fault/failure insertion in the software and hardware. Another area of interest is that of electromagnetic interference.

3.6.2 V&V of Procured Equipment

Should BRR purchase equipment from another manufacturer, the manufacturer must provide a "safety case" with full supporting documentation. This should include evidence that the manufacturer has taken into consideration appropriate safety and quality procedures such as those in RIA 23. A Safety Assessment Group that comprises various BRR experts and perhaps an independent organization, will conduct a type of review or audit on the evidence in order to recommend approval for use of the equipment. Each business within the British Rail system (e.g., Intercity) will give final approval based on the recommendation by a "safety panel" of signalling experts.

3.6.3 RIA Technical Specification No. 23

This document, prepared for the railway signalling industry in the United Kingdom, describes procedures and technical requirements for the development of programmable electronic systems used in safety critical applications. As indicated earlier, its primary intent is to define how the IEC draft standard 65A (Sec.) 122 is to be used in the railway industry. For this reason, its emphasis is on software aspects. It is expected that this document will be expanded at a later date to address system/hardware design issues.

RIA 23 is structured around three main concepts: integrity levels, roles/responsibilities of individuals and organizations, and life cycle models.

3.6.3.1 Integrity Levels - The concept of safety integrity level (criticality or importance of safety) is discussed because various activities and techniques are recommended (later in the document) for the various levels.

3.6.3.2 Roles/Responsibilities - Roles and responsibilities are defined for four main parties:

- Safety authority -- party responsible for certifying/approving that the system is fit for service
- Design Authority -- party supplying the system; responsible for verification activities
- Validator -- party responsible for validation activities

- Assessor — party performing an auditing role to ensure that the Design Authority and Validator have selected and properly applied appropriate V&V techniques.

3.6.3.3 Life Cycle Models - Activities/techniques relative to three main hierarchical life cycle models are described: system life cycle, development life cycle and cross-life cycle. Areas addressed in the system life cycle include requirements definition/analysis; system design specification; hardware and software development; system acceptance and commissioning; and monitoring, maintenance and revision. The development life cycle includes: software architectural design, detailed software design, software module coding, software integration and system integration. The cross-life cycle includes: configuration management, quality assurance, verification and testing, validation, assessment (audit), and documentation. It is the verification/testing and validation areas that are addressed in more detail below.

3.6.3.3.1 Verification and Testing - Verification is defined as the process of establishing, at each stage in the development process, that the system is compliant with the requirements specified by the preceding stage. Testing is a key part of this process. One table in Appendix A of the specification provides guidance on selecting the various verification and testing techniques to be applied, based upon the assigned safety integrity level. One of five recommendations is given for each possible integrity level: mandatory, highly recommended, useful, not recommended or forbidden. Another table provides a testing "checklist." Verification and testing are to occur at various stages throughout the development process. Examples of techniques include simulation, cause-effect diagrams, FMECAs, Markov models, formal design reviews, static code analysis and symbolic execution; thirty techniques are listed.

Requirements for documenting results of the verification and testing activities are also presented.

3.6.3.3.2 Validation - Validation involves demonstrating to the Safety Authority that the system is fit for use. It involves activities conducted by the Validator throughout the development life cycle, concluding with the system acceptance and commissioning phase. One of the five recommendations described above is given by integrity level for each of five different possible validation techniques: check-lists, Fagan inspections, formal design reviews, reliability growth model and rigorous correctness arguments. In addition, the Validator is urged to use additional verification and testing techniques as deemed necessary.

3.7 INTERNATIONAL UNION OF RAILWAYS

The Union International de Chemin de Fer (UIC), or International Union of Railways, is located in Paris, France, and has been involved in the standardization of railway signalling in Europe for several years. It comprises several working groups and subcommittees whose roles have included setting standards and making recommendations for its member railway organizations. The UIC does not set required levels of safety; this is left to the member railways. The UIC also does not conduct safety verifications/validations; these are the responsibility of the equipment supplier and sometimes the user railway itself.

An important research arm for the UIC is the European Railway Research Institute (ERRI), formerly the Office des Recherches et d'Essais (ORE), or Office of Research and Experiments. It is located in Utrecht, Netherlands. The role of the ERRI (ORE) has been to conduct research on behalf of its members and publish results (typically recommendations). These results then serve as the basis for standards, which are the responsibility of the UIC.

In 1969, two special committees were created within the ORE to study various issues pertaining to the "use of electronics in railway signalling." One committee, namely A 118, was charged with looking into the use of electronics in signalling in general. Committee A 155 was charged with looking more into the use of computers and the transmission of safety-related information. Work of these two committees culminated in the production of approximately 30 separate reports, dealing with a wide range of safety-related issues from specifications, design and proving of safety, to documentation and management requirements.

In 1990, the UIC published the document "Processing and Transmission of Safety Information," UIC 738 R (2nd Edition, dated 1/1/90). This document summarizes the work of the ORE committees, and is the latest effort of the UIC in this area. UIC 738 R was not meant to be a compulsory standard, but rather a recommendation (for railways and suppliers) for the specifying, designing, validating, applying and modifying of safety systems in railway applications. The document includes a general description of a recommended process to validate the safety of computer-based railway signalling systems. It is this document, together with several of the ORE reports, which provide the basis for the V&V process recommended by the UIC.

There are, in fact, a number of organizations in Europe which are involved in establishing standards in railway signalling, including those pertaining to the verification and validation of safety critical equipment. The structure/interaction of these organizations is quite complex. However, it should be noted here that the politically-based European Community (EC) is in the process of setting V&V and other standards through the Directorate General Transport Commission (DG VII) and two other associated organizations referred to as CEN and CENELEC. Work in the area of railway signalling/communication is being performed by working groups, technical committees (e.g., TC 9X) and subcommittees (e.g., SC 9XA) in the CENELEC organization. Another committee, the Joint Railway Programming Committee (JPC RAIL), is monitoring the work of CEN and CENELEC and is responsible for setting priorities for standardization. Still another organization, the Community of European Railways (CER), acts as an intermediary between the UIC and the European Community.

The EC (and in particular, the TC 9X and SC 9XA groups within CENELEC) intends to release two V&V standardization documents (currently in draft form) sometime in 1994. One is to pertain to software safety and the other to hardware and system safety. These are to describe the procedures which are to serve as standards for the "proof of safety" of computer-based railway equipment for EC member organizations. At the present time, the CEN/CENELEC organizations have adopted the existing UIC recommendations until the new standards are released. These two draft standards are discussed in more detail in the section addressing the German Federal Railway.

3.7.1 Safety V&V Process

The process recommended by the UIC to validate the safety of computer-based railway equipment is described in general terms in UIC 738 R, and in more detail in the following two specific ORE documents:

- "Question A 155, Use of Electronics in Railway Signalling, Report RP 11 - Proof of Safety of Computer-Based Safety Systems," dated September 1987

- "Question A 155.1, Transmission of Safety Information, Report RP 8 - On Proving the Safety of Transmission Systems," dated April 1986.

The former ORE document (i.e., RP 11) provides more detailed information on the various methods and steps which can be used in the validation of computer-based systems. It covers both hardware and software with emphasis on the software. The latter ORE report (i.e., RP 8) deals with both qualitative and quantitative methods for proving the safety of information transmission systems. The V&V process described below is based primarily upon these three documents. It should be emphasized that this process is a recommendation which cites different methods that can be used.

Validation, in these documents, refers to the process by which the safety of a new or existing system or subsystem is established. The complete set of all validation documentation comprises the proof of safety. The process is be performed by someone different from the specifier, designer or constructor of the equipment -- usually a separate group within the manufacturer's company or the railway itself. An independent safety audit is also recommended to assure that all validation processes have been carried out and documented correctly.

The validation process itself is based upon a collection of analyses, calculations, simulations and tests. Each of these complement one another, and it is the results of these taken as a whole which provide the confidence in the safety of the equipment. Both qualitative and quantitative analyses are recommended, with each complementing the other; however, qualitative aspects are considered most important. The process applies to all types of safety critical railway equipment including on-board and ground-based systems. It also covers the entire life of the system from specifications to post-installation modifications.

The validation process is composed of three main aspects: software validation, hardware validation and system validation. Another aspect, namely, validation of information transmission systems, involves all three of these areas. More details on these four aspects are provided on the following pages.

3.7.1.1 Software Validation - Validation of the software consists of verifying that it performs the functions required by the specifications for all internal and external variables. This not only includes the object code of the program itself (e.g., application software, operating/executive system software), but also any higher level language which was produced by development software (e.g., assembler, compiler). Thus, the process must show that the object code was correctly produced by the compiler/assembler.

In order to facilitate the validation process, the software should be decomposed (during the design) into separate modules or sub-programs which can be validated separately, and then in an integrated manner. The software validation described here pertains to modules and complete software systems which have already been designed. Design aspects relating to software are addressed in the system validation section.

Software validation is based upon the conduct of qualitative and quantitative analytical methods as well as testing. These are described below along with other validation concerns/ aspects (e.g., assembler/compiler validation, data validation).

3.7.1.1.1 Analyses - A number of analytical methods are recommended for the software. Some or all of these could be performed depending upon the complexity of the software program and the desired extent of the validation:

- Program inspection - Reading the code and looking for common programming errors; good for simple software programs

- Program walkthrough - Group of experts discuss the program line-by-line

- Program analysis - Function of the program is derived or reviewed on the basis of machine code instructions to 1) check agreement between program and specification, and 2) provide a basis for meaningful tests based on program structure

- Syntactic or structural checks - Check for flaws in the construction/basic structure of the program

- Programming language syntax checks - Check of syntax of programming language

- Control flow analysis - Graphical technique to detect errors such as unreachable code or dead-end branches

- Data or information flow analysis - Check for use of undefined variables

- Semantic analysis - Analysis of program semantics (i.e. functional interaction of program modules and meaning of program variables)

- Symbolic execution - Executing program with symbols for the input variables; results in large amounts of data

- Formal proof - Mathematically prove the correctness of the program; difficult to implement for complex programs.

3.7.1.1.2 Testing - Testing is to be performed on the individual modules as well as the entire software. It can be carried out in a simulated environment or on the target hardware itself. Recommended tests include the following:

- Tests according to requirement specification - Software is tested independently of its structure via set of input data; check for unspecified results

- Tests according to program structure - Apply set of input data which requires execution of program branches/paths; particularly good to do on individual modules as well as overall system

- Testing against a diverse implementation - Compare output of program with that of another program based on same specifications.

3.7.1.1.3 Quantitative Methods - Two approaches are possible here to quantify software quality or reliability. One involves examining the software relative to specific quality criteria, and generating an associated quality index. The other involves estimating software reliability via mathematical models and the number/type of errors detected during previous/analyses testing. Although these do not provide absolute results, they can provide additional confidence in the lack of latent software errors in the system.

3.7.1.1.4 Assemblers/Compilers - Assemblers (especially macro-assemblers) and compilers, and in particular, the resulting software thereof, should be validated in the same way as the main programs.

3.7.1.1.5 Data Validation - Data in data-based software programs should also be validated. The method here is similar to that described in the document "Techniques for Verification and Validation of Safety Related Software," for the European Workshop on Industrial Computer Systems (EWICS) TC 7, Working Paper 267/4, August 1983. This document was not available for review at the time of this report.

3.7.1.1.6 Temporal Validation - This involves checking the software timing characteristics against what may be expected in the actual operating environment. Program execution times are checked as well as checks directed to obtaining correct software responses based on different input conditions. Logic analyzers can be used to assist in this process.

3.7.1.1.7 Special Validation for Systems with Diverse Software - When two or more software programs, or portions thereof, perform the same functions, special analysis and testing techniques are needed in the validation process; these are directed to the following:

- Checking a system's ability to detect and safeguard against low level programming errors - check to see if programs are free from errors that yield congruent results, or if an error in one program yields a different error in the other program(s)

- Checking a systems' ability to detect and safeguard against high level programming errors - check to see if design/programming team(s) do not misinterpret specifications

- Checking a system's ability to detect and safeguard against hardware failures - check to see if programs react differently to similar hardware failures, and that the output differences are detected in a safe manner.

3.7.1.2 Hardware Validation - Validation of the hardware consists of checking for its safe functioning under normal operation (without hardware failures) as well as under failure conditions. Both analytical and testing methods are recommended. Failure modes and effects analyses should be used on non-redundant safety critical hardware. Component failures which should be considered are identified in the ORE report, A 155.3 - "Failure Catalogue for Electronic Components," Report RP 12, April 1988. A fault tree analysis is another recommended technique. Both of these techniques can be used for qualitative as well as quantitative analyses, but qualitative is of primary importance. It should be noted that validation of "highly integrated circuits" has not been resolved in the process described.

Laboratory tests are recommended to complement the analyses of the hardware. The tests should be used to check hardware operation over the full range of environmental conditions, especially temperature, power supply voltage variances and electromagnetic disturbances. The tests should also check for the proper functioning of protection mechanisms. This could be qualitative and/or quantitative. Injection testing or simulating hardware failures is recommended for qualitative testing; this is limited to single failures. Quantitative testing involves injection testing and interpreting the results on a statistical basis.

3.7.1.3 Transmission System Validation - Additional activities are needed to validate computer-based information transmission systems -- those system portions which are involved in the encoding, transmission and decoding of safety critical information. In many instances, due to the nondeterministic nature of error sources/distributions and quantitative safety requirements, it is necessary for this validation to include quantitative methods. Complete validation of such system portions is to begin with qualitative analyses, followed by quantitative techniques, and then testing.

Qualitative validation is to be based upon FMEA and fault tree techniques, which are considered complementary. The primary interest is the identification of potentially unsafe hardware failures, software errors and conditions associated with human errors. Special attention is directed to such aspects as the mathematical formula used in coding, synchronization of transmission channels, loss of messages, source/destination addresses, timing, cross-talk and the use of public networks (if applicable). Another qualitative aspect is an activity referred to as "protocol verification." This is where the communication protocols are checked to see if lower level errors are detected by higher levels in the protocol. These are based upon three methods referred to as reachability analysis, program proving and error-free construction.

Quantitative methods based on system modelling are to follow qualitative analyses. These are to be based upon either calculation or simulation techniques which are directed to quantifying the probability of errors in the transmission medium. Simulation is recommended over calculation for very large and complex systems.

Testing is then to be performed to support the qualitative and quantitative methods. This is to consist of laboratory tests and field tests/trials.

It is expected that some combination of the methods discussed above should be used based upon the system design and error sources.

3.7.1.4 System Validation - Although the software, hardware and transmission systems are to be validated separately, the overall system must also be validated. This is done through a series of analytical checks, bench testing and on-site testing. It is considered most effective if the validator can be involved during the actual system development effort, but post-development system validation is also possible.

3.7.1.4.1 Design/Development Activities - A number of activities are recommended during the design/development stage of the system to minimize the risk of validation failure. These are directed to the design process, and reference a number of the ORE reports that deal with design issues of both hardware and software. Note that these are not actually part of the validation process itself. The recommended activities include the following:

- Verification of each of the levels of specification and validation of the transition from one level to another

- Justification and recording of the design choices made and solutions dismissed

- Description of the nature and intent of the protection mechanisms utilized

- Observing design recommendations in ORE reports A 155/RP 7 and RP 9

- Producing clear and accurate documentation

- Correct structuring, simplicity, legibility and maintainability of the software
- Observing software design guidelines in Report A 155 RP 9
- Evaluation of the proposed hardware structure and effectiveness of the protection mechanisms, perhaps by Markov modeling
- Observing hardware design guidelines in Report A 155 RP 7.

3.7.1.4.2 Analyses - System validation begins with a fault tree analysis on the entire system. Following this, a system level failures modes and effects analysis is recommended to address the effects of hardware failures on the execution of the software.

3.7.1.4.3 Bench Testing - Bench testing is recommended to determine that the final hardware and software operates together as intended; this consists of two parts. First, the software of each element of the system is tested. Then, the system is tested with the software assumed to be free of errors. The tests should consist of the following:

- Tests employing known data (prepared manually or automatically) and comparing outputs of individual software elements with expected results
- Tests using simulators in real or accelerated time to check functioning of the system (hardware) as a whole over extreme values of the specification
- Tests to ensure that all parts of the system are fully exercised, particularly start-up and power failure procedures
- Tests to prove the effectiveness of error detection mechanisms by injecting failures into the system
- Tests to prove the correct functioning of data links in presence of communication failure and excessive data error rates.

3.7.1.4.4 On-Site Testing - Site testing is recommended in order to ensure that the system performs as intended in its actual application and environment under real input conditions.

This type of testing is also recommended for post-installation modifications, which are not addressed in great detail by the existing documentation.

3.8 TÜV RHEINLAND

Transportation systems in Germany must be certified prior to revenue service operation by the local government body or supervising authority. Typically, an independent assessor organization such as TÜV Rheinland (Cologne, Germany) examines "proof-of-safety" documentation submitted by the developer and provides a recommendation to the local governing authority as to whether or not to accept the system. TÜV Rheinland has had this assessor role on the Dortmund H-Bahn, and is currently involved in the Frankfurt Airport Passenger Transport System (PTS) and the Transrapid maglev system at the Emsland Test Track. The German Federal Railway (DB) has the distinct opportunity of being its own assessor and certifier.

A key aspect of this certification process (involving an independent assessor such as TÜV) is the establishment of an agreement between the developer and assessor in the early stages of a project as to what safety measures (activities and associated documentation) are to be undertaken. The developer proposes a "bundle" or set of measures which comprises their proof-of-safety, and this is then approved and/or revised by the assessor. This includes all safety verification and validation issues. The primary basis in the area of computer system safety for the measures proposed by the developer and accepted by the assessor is the following set of four German standards:

- DIN V VDE 0801, "Principles for Computers in Safety Related Systems"
- DIN V 19250, "Fundamental Safety Analyses for MSR (Measurement-Control-Regulation) Protective Devices"
- DIN VDE 0831, "Electrical Equipment for Railway Signalling"
- Mü 8004, "Principles for Technical Approval for Signalling and Communications Technology."

The first three are published either separately or jointly by the DIN (German National Standards Institute) and/or VDE (German Association of Electrical Engineers -- similar to IEEE in the U.S.), while the latter document is produced by the Deutsche Bundesbahn (DB) -- German Federal Railway. Further, while DIN V VDE 0801 and Mü 8004 are railway-specific, DIN V 19250 and DIN VDE 0831 are more generic in nature. The certification agreements are usually based more on DIN V VDE 0801 and Mü 8004. In the case of the DB, Mü 8004 is the primary document. Brief overviews of these standards (except for Mü 8004, which is addressed elsewhere in the German Federal Railway section of this report) are provided later in this section.

There are also other applicable documents (prepared either totally or in part by TÜV Rheinland staff) which provide additional guidance on the design and/or assessment of computer-based safety critical systems. Those include:

- "Minimum Requirements for Safety Related Computers in Railroad and Nuclear Engineering," (TÜV Rheinland and TÜV Norddeutschland, 1988)

- "Microcomputers in Safety Technique-An Aid to Orientation for Developer and Manufacturer," (TÜV Rheinland and TÜV Bayern, 1986)

- "TC7: Systems Reliability, Safety and Security," (TÜV Rheinland, 1985).

These documents, while not standards, provide more detail for the developer and/or assessor on the safety measures identified in the above mentioned standards. Brief overviews are also provided for these three applicable documents, following the overviews of the standards.

3.8.1 TÜV's Safety Assessment Process

As indicated above, TÜV Rheinland acts as the assessor in examining the proof-of-safety provided by the developer. In earlier projects (e.g., Dortmund H-Bahn), TÜV's involvement began after the development of the system and after the documentation was generated. However, in more recent systems, involvement has occurred in parallel with the development process. It should be noted that the involvement is still heaviest toward the end of the development process so as to minimize interference with the developer. It should also be emphasized, however, that the agreement on what to be done is established early in system development.

On at least one of TÜV's recent projects (i.e., the Frankfurt Airport transportation system), a detailed Certification Plan and documentation database has been established to track the overall certification process. This plan is referred to as the Project Accompanying Safety Certification (PASC).

The actual assessment process used by TÜV Rheinland to help ensure the safety of computer-based railroad applications is essentially project dependent. The primary reason for this is that the safety measures as selected from the various standards and agreed upon by the developer and assessor (TÜV) are also project dependent. However, the assessment involves safety verification and validation of the computer hardware and software at different times throughout development, and involves a combination of reviews, inspections, analyses and tests. Due to increasing complexity in software, emphasis is shifting more from analyses to testing. Also, much of the burden of testing is placed upon the developer. TÜV's role (after required tests are initially agreed upon) is to observe testing at the developer's facilitates, but also to conduct additional tests if deemed necessary. Also, TÜV takes into account, as much as possible, the analyses performed by the developer or by a third party for the developer.

The following two documents provide some insight into the process used by TÜV Rheinland to assess the safety of (essentially) the software of computer-based systems:

- SBT 90.01/00/E "Guidelines for the Assessment of Safety-Relevant Computer-Systems in Railroad Technology," and

- WP 520, "Verification of Safety Related Programs for a Maglev System."

The former document is more of an informational document for customers on the certification process, while the latter describes in more detail the process used to assess the software of the first generation maglev system at the Emsland Test Site in Germany. The current process used by TÜV Rheinland for assessing software is generally similar to that described in the second document. Their assessment process for the hardware is not in formal documentation.

Below are provided brief overviews of key portions of these two TÜV documents followed by the remaining overviews as described above.

3.8.1.1 Guidelines for the Assessment of Safety Relevant Computer Systems in Railroad Technology - Following a general description of the assessor's involvement with the developer in the certification process, a basic software verification and validation approach is discussed. It generally involves the following steps:

- Validation of system requirements
- Assessment of the software requirement specification
- Assessment of the software architecture (software design-first level)
- Assessment of the software design at the nth level
- Analysis at the source code level
- Test of the machine code
- Test of the software
- Integration test (hardware and software) .
- Review of operating instructions
- Safety trial run
- Preparation of final safety report.

Again, this process is very general in nature. The nature and timing of specific reviews, inspections, analyses, and tests are decided upon in the early design phases and documented in the proof-of-safety agreement.

This document also provides some highlights from DIN V VDE 0801 as to what constructive and analytical measures (design and assessment) can be used by the developer to reduce software errors.

3.8.1.2 WP 520, Verification of Safety-Related Programs for a Maglev System - This document describes the verification procedure used by TÜV Rheinland to examine the software in the first generation Transrapid maglev safety system at Emsland. The verification approach was based upon a combination of analyses and tests, and a scheme referred to as "diverse back (reverse) analysis." As mentioned earlier, a process very similar to this is currently being used in other assessments. However, complete reverse analysis is not utilized in more complex applications.

The basic scheme involves a step-by-step redevelopment of the specification beginning with the previously developed code in EPROM, and then a comparison of the redeveloped specification with the original specification used in the development of the software. The initial process involves translating the existing machine code into mnemonic code, and then into two different translations of the higher order language, Pascal. A special discompiler tool referred to as DISCO is used for the translation. Then, even higher level statements are manually generated from the Pascal translations using flow chart techniques. Problem statements are developed from these higher level statements and compared with the original specification. The objective of this entire process is "software verification" — just one aspect of assessing system safety. Validation is defined in this document as the process in which the conformity between the software and the requirements of the overall computer system is established. The relationship between verification and validation is addressed further in the previously discussed document.

3.8.2 Applicable Standards

Overviews of the four applicable standards from which safety measures are chosen and agreed upon by the system developer and assessor are provided below.

3.8.2.1 DIN V VDE 0801 - This preliminary standard (as indicated by the "V" preceding VDE), entitled "Principles for Computers in Safety-Related Systems," is a quite extensive document (over 170 pages) that applies to both the hardware and software of computer systems in safety critical applications. It is the primary safety standard currently being used in railway applications in Germany; the exception is the DB, which uses Mü 8004 almost exclusively. The primary focus of the document is the presentation of various safety measures that can be used throughout the entire life cycle of a computer system to avoid and/or control errors, where an error is defined as an undesired system response.

Measures to avoid errors are presented in Chapter 5, and are separated into the following system life cycle phases: conceptual, development, manufacturing preparation, manufacturing, installation, operation and modifications. One key phase of interest here is development, in which the measures are further divided into technical and organizational measures. The technical measures are then divided even further into constructive and analytical measures that can be taken during both hardware and software development to avoid/detect errors. Examples of analytical measures during hardware development include statistical analysis, dynamic analysis, simulation, break effect analysis, and functional analysis. Examples of analytical measures during software development include inspections, walkthroughs, symbolic design, and systematic tests (white or black box). Examples of constructive measures during software development are use of computer-aided drafting systems, modularization, structured programming and use of more sophisticated languages.

Another area of interest is that of modifications, and the measures that can be used to avoid/detect errors. Steps are described for formalization of the modifications. Recommendations include conducting an analysis of the consequences of the modifications and an operational checkout of the modified system. However, it is acknowledged that a complete repetition of all previously conducted tests and checkouts is usually not necessary.

Measures to control errors are presented in Chapter 6, and are divided into three groups:

- Structural measures at the system level (e.g., use of redundancy, self test, monitoring techniques)
- Measures below the system level (e.g., specific memory ranges)
- Non-technical measures (e.g., personnel qualifications, training).

Additional descriptions of all measures are provided in the appendices of the document.

A procedure is also presented on how to select the appropriate measures based upon the requirements classes (risk classifications) of the system. More information on determining risk classifications is provided in the standard DIN V 19250. As part of this procedure, all measures to avoid and control errors in the system are separated into three groups as follows:

- Group I, Basic measures – must be used regardless of the requirement class
- Group II, Non-replaceable measures – must be used regardless of the requirement class, but can be staged
- Group III, Replaceable measures – can be replaced individually or in connection with other measures.

Then, guidance is provided on how to select these measures.

3.8.2.2 DIN V 19250 - This document, "Fundamental Safety Analyses for MSR (Measurement-Control-Regulation) Protective Devices," describes a procedure for conducting safety analyses of MSR devices – those portions of a system performing protective or safety critical functions. The procedure is actually a qualitative risk assessment that leads to the identification of requirement classes for the protective device. These requirement classes (a total of eight are possible) can then be used in conjunction with DIN V VDE 0801 to help in the selection of safety measures to be used by the developer and/or assessor. The document is generic in nature and applies to the railway industry as well as others.

3.8.2.3 DIN VDE 0831 - The June 1983 version of this standard, "Electrical Equipment for Railway Signalling," primarily describes design and installation requirements/specifications for railway signalling systems (e.g., selection/location/installation of materials, wiring, signal lamps, point machines). The document was apparently not developed to address specifics of computer-based systems. A later version, published in 1990 but not available at the time of this report, addresses related requirements for computer-based systems.

The available document at the time of this report (June 1983 version) does not specifically address safety verification or validation processes or techniques. However, Section 6 describes some basic safety requirements for signalling equipment. In particular, Sections 6.2.2 and 6.2.3 indicate that a fault condition should be prevented and that a single fault should not lead to an impermissible fault condition -- one that endangers railway operations. These sections generally describe the need to detect faults and ensure a safe response.

A section entitled Proof of Safety (i.e., Section 6.4) merely indicates that checks should be made to ensure compliance with this standard.

Section 8 provides only a brief and general discussion of acceptance and modification testing. It specifies that three types of testing are necessary: general testing, functional testing, and circuit testing. Modification testing is to be carried out in the same manner as acceptance testing.

3.8.2.4 Mü 8004 - As indicated earlier, Mü 8004 is a German Federal Railway (DB) standard, and is described in that section of this report.

3.8.3 <u>Other Applicable Documents</u>

Several other documents (in which TÜV Rheinland played a role in producing) that provide guidelines/insight into the design/assessment of computer-based systems are discussed below.

3.8.3.1 Minimum Requirements for Safety-Related Computers in Railroad and Nuclear Engineering - This research report was prepared jointly by TÜV Rheinland and TÜV Norddeutscheland in order to develop some commonality and general agreement between safety requirements for the railway and nuclear industries. It discusses basic design and assessment requirements for the hardware and software of computer-based systems in safety critical applications. An English version of a portion of the document pertaining to software was available for this project. This version, entitled "Possibilities for Design and Testing of Fail-safe Computer Systems (Software)," discusses design and assessment methods for safe software.

The focus of assessment in this software document is on verification which is to show that the software is both correct and valid and will not result in any unsafe system states. This process is dependent upon the criticality of the application and the system structure (e.g., single channel or diverse software), and is to entail both analyses and testing. Chapter 5 of the document discusses the attributes of the various analysis and test methods that could be used. Included are techniques such as black box tests, systematic tests, code inspections, walkthroughs, and manual program analysis. The remainder of the document discusses types of software diversity and presents an example of one software verification method.

According to this report, the three basic methods for producing safe software are: 1) using measures to design reliable software, 2) applying quality assurance procedures, and 3) conducting software testing (and analyses).

3.8.3.2 Microcomputers in Safety Technique - This research report, produced jointly by TÜV Rheinland and TÜV Bayern, presents a catalogue of safety measures for computer-based systems and a means of selecting appropriate measures for a given application. A large number of design and assessment measures are discussed including single channel and multiple channel structures, comparators, checksums, RAM/ROM tests and monitoring, input/output tests, diversification of software/hardware, code inspection, manual program analysis and diversified reverse analysis. A form of the latter technique is currently used by TÜV Rheinland in the verification of software. The intent of the report is to provide guidance on the safe and cost effective construction of computer-based systems.

3.8.3.3 TC7: Systems Reliability, Safety and Security - This report, prepared by TÜV Rheinland in conjunction with the European Workshop on Industrial Computer Systems in 1985, presents guidelines on the development and associated documentation of safety-related software. Its intent is to provide guidance on developing software that is as error free as possible. It does not address the proving or verification of the software program, nor does it cover hardware considerations.

3.9 GERMAN FEDERAL RAILWAY

The German Federal Railway (DB) is responsible for defining safety requirements/standards for signalling and train control equipment and approving the use of such equipment on its railways. The primary division of the DB which has these responsibilities is the Railway Central Office or Bundesbahn Zentralamt (BZA), located in Munich, Germany. The specific role of this (independent) office is to interact with the manufacturer in the development of the equipment, perform safety verifications and validations (including some proof-of-safety testing), and to approve the equipment for use by the DB. It must ensure that all safety requirements, which are applicable throughout the entire development cycle, are met. Manufacturers must demonstrate compliance with all such requirements.

The primary "standard" which defines these requirements for computer-based and conventional systems is the DB document, Mü 8004, "Principles of Technical Approval for Signalling and Communications Technology." This document describes safety requirements applicable to the development of the equipment, and describes the structure and content of a proof-of-safety document which must be provided by the manufacturer. The proof-of-safety document must prove in verifiable form that all safety rules/requirements have been observed. A brief overview of Mü 8004 is provided below.

Another standard which also applies is DIN VDE 0831, "Electrical Equipment for Railway Signalling." This is addressed in more detail in the section on TÜV Rhineland.

In addition to safety requirements, a separate requirements specification for functional aspects is jointly prepared by the manufacturer and the DB, and is closely reviewed by BZA for completeness and correctness. This document becomes a precondition for the functional specification and system design.

It should also be noted that the manufacturer must also meet quality assurance requirements such as those described in the European Norm 29000 (or ISO 9000) series of standards. Evidence of meeting quality assurance requirements sometimes includes a compliance statement from an independent organization who has examined quality assurance procedures in the company of interest over a period of time. In other instances, BZA may conduct an inspection of a manufacturer's quality assurance process.

Of key significance to the DB as well as other European railways is the effort underway via CENELEC (described more in the section on UIC) to develop common safety standards for all Railway Administrations in Europe. Two working groups (i.e. WGA1 and WGA2) have been established to develop these standards, and the documents being prepared are as follows:

- WGA1 – "Railway Application: Software for Railway Control and Railway Protection Systems," and

- WGA2 – "Railway Application: Safety Related Electronic Railway Control and Railway Protection Systems."

The former addresses software, while the latter covers system/hardware related issues. At the time of this report, these documents were in the fifth or sixth draft stage. One basis for the content of these drafts is Mü 8004. Another major source of information for these new standards is the following set of International Electrotechnical Commission (IEC) standards which were identified earlier in this report:

- IEC 65A (Secretariat) 122, "Software for Computers in the Application of Industrial Safety Related Systems," and

- IEC 65A (Secretariat) 123, "Functional Safety of Programmable Electronic Systems: Generic Aspects."

A copy of the fifth working draft of the WGA2 standard for system/hardware aspects was obtained for this study. A brief overview of its nature/content is provided below, following an overview of Mü 8004. A copy of the WGA1 draft related to software was not available at the time of this report. However, indications from the BZA in Munich are that the WGA1 draft will be very similar in content to the IEC software document IEC 65A (Secretariat) 122. This IEC standard is addressed in more detail in the IEC section of this report.

3.9.1 Mü 8004

Mü 8004, "Principles of Technical Approval for Signalling and Communications Technology," comprises the following major sections (chapters):

- Chapter 0 — References and Table of Contents
- Chapter 1 — Technical Approval for Materials of Signalling and Communications Technology
- Chapter 2 — Requirements Specification
- Chapter 3 — General Guidelines for Vital Circuits and Equipment of Electronic and Relay Technology
- Chapter 4 — General Guidelines for Vital Circuits and Electronic Equipment
- Chapter 5 — General Guidelines for Vital Circuits and Relay Equipment
- Chapter 6 — Guidelines/Basic Principles
- Chapter 7 — (left blank)
- Chapter 8 — Guidelines for Technical Documentation
- Chapter 9 — Information Concerning These Guidelines.

3.9.1.1 Chapter 4 (Safety Analysis) - Chapter 4, entitled "Safety Analysis," is the primary portion of the document of interest in this study. It essentially describes the requirements and nature of the safety analysis or proof-of-safety activity to be conducted (primarily) by the manufacturer. The purpose of this activity is to show that the equipment is "fail-safe" according to the guidance provided in Chapter 3 of the document. This safety analysis will subsequently be "validated" by the BZA, who also will jointly be involved in some of the safety analysis activities such as testing.

The safety analysis section is subdivided into several major subsections as follows:

- 41000 — Preliminary Remarks
- 42000 — Functional Verification
- 43000 — Failure Effects
- 44000 — Effects of Electrical Interferences
- 45000 — Safety Specific Application Regulations.

The latter four subsections are addressed below.

3.9.1.1.1 Functional Verification (42000) - In general, this activity is to show that the operational requirements (from the functional specifications) and safety requirements are met under failure-free and fault-free conditions. Major areas of requirements, covered primarily in Section 42500 (Computer Programs), are listed below:

- Functional Description – software and aspects thereof (e.g., functions, interfaces, interrupts, data flow) are to be described in such a way so as to facilitate checking for compliance with safety requirements

- Structure – software is to be structured so that functions performed are evident

- Program Flow – one of three program structures is allowed: sequence, selection or repetition

- Program Module – modules are to represent unique and distinct subfunctions

- Coding – guidance is provided on coding aspects including programming language, loops, calculations, subroutine returns, constants, etc.

3.9.1.1.2 Failure Effects (43000) - This task is to show that failures of functional units result in safe states. It is divided into five aspects as follows:

- Single failure – examine effects of single component failures; an extensive failure mode list is provided

- Independence – demonstrate that two or more functional units (in which a simultaneous failure in each could be unsafe) are independent (e.g., coupling, cross-talk)

- Detection of failures – show that (potentially unsafe) failures are detected within an appropriate time and lead to safe states; special tests may be needed for integrated circuits; various tests are needed for the software (e.g., command check, ROM and RAM checks)

- Multiple failures – may need to show effects of certain sequential and/or multiple failures (e.g., one failure not detected in a timely manner); up to three failures may need to be checked

- Safe state – show that all detected failures lead to a safe state.

3.9.1.1.3 Effects of Electrical Interferences (44000) - This task involves showing that electrical interferences do not have any unsafe effects on system operation.

3.9.1.1.4 Safety Specific Application Regulations (45000) - This task involves ensuring that safety-related regulations are properly addressed in the planning, construction, operation and maintenance of the installation.

3.9.1.2 Mü 8004 Supplements - Mü 8004 is a working document within the DB, and periodically receives revisions and supplements for specific applications/purposes. Some of the key supplements which have been included in Mü 8004 (primarily in Chapter 6) are as follows:

- "Guideline with Supplementary Specifications for Testing Software in PASCAL"
- "Basic Principles for the Application of the Problem-Oriented Program Language PASCAL in Vital Railway Signalling Equipment"
- "Guidelines with Supplementary Specifications for Testing Software in PASCAL"
- "Guideline for Testing Software of Fail-Safe Installations"
- "Basic Principles for the Approval of PASCAL Compilers Used for the Translation of Programs for Vital Railway Signalling Equipment"
- "Basic Principles for a Safe Indication of Information on Display Equipment Used with Electronic Interlockings"
- "Guideline for Conditions to be Met in the Safe Transmission of Information for Railway Signalling Equipment"
- "Guideline for Testing the Software of the SIMIS 3116 and SIMIS 3216 Computers."

3.9.2 WGA2 CENELEC Draft Standard

This draft standard (CLC/TC9X/SC9XA/WGA2, version 5, dated April 1993), being prepared by working group WGA2 of the CENELEC organization, pertains to the safety of system/hardware aspects in railway electronic control systems. As mentioned earlier, it is an attempt to develop common safety standards for railways within the European Community. Once completed, it is intended to be used in conjunction with the software standard being prepared by working group WGA1. This WGA2 draft, which applies specifically to railway control and protection systems, defines the requirements or conditions which must be satisfied in order to be accepted as "adequately safe" for its intended application. It applies to the specification, design, construction, installation, acceptance, operation, maintenance and modification/extension of complete systems as well as individual subsystems/components. It does not deal with occupational health and safety issues.

Requirements for safety acceptance are separated into three main areas: evidence of quality management, evidence of safety management, and evidence of technical safety. The documentation evidence in these areas is to be presented as a "proof-of-safety" for generic systems/subsystems and a "safety case" for specific applications.

3.9.2.1 Evidence of Quality Management - Overall quality of the system/equipment is to be controlled via an appropriate management process throughout its life cycle. The quality system/process is to comply with ISO 9001/EN29001, and the organization should obtain ISO 9001 certification. The purpose of this is to minimize the risk of certain errors/failures in the system/equipment.

3.9.2.2 Evidence of Safety Management - A safety management process is to be utilized throughout the life cycle. This process is to include the establishment and implementation of a safety organization, overall safety plan (e.g., activities, milestones), hazard log, safety requirements specification (including a hazard analysis, risk assessment and safety level assignment), structured design methodology, safety review plan, and safety verification and validation plan. The latter is to ensure that each phase of the life cycle satisfies safety requirements of the previous phase (i.e., verification), and the completed system satisfies the original safety requirements (i.e., validation).

3.9.2.3 Evidence of Technical Safety - Evidence of safety of the design is to be provided in a Technical Safety Assurance Report, forming a portion of the overall proof-of-safety or safety case documentation. This report is to include results of all activities (e.g., analyses, testing) which contribute to showing the safety of the design.

Four or five topics are to be addressed:

- Assurance of correct operation — show correct operation occurs with no faults in existence; should address:
    - Definition of interfaces
    - Fulfillment of functional requirements
    - Software/hardware interaction (must comply with WGA1 software standard)
    - Assurance of correct hardware
    - Fulfillment of environmental conditions
    - Presentation of software functions and proof of correctness (must comply with WGA1 software standard).

- Effects of faults — show that system/equipment meets safety requirements in the presence of hardware and systematic faults; should address:

    - Independence of items (where appropriate)
    - Effects of single faults (via e.g., FMECA)
    - Detection of single faults
    - Safe action following detection
    - Effects of multiple faults (via e.g., FTA)
    - Protection against systematic faults.

- Operation with external influences — show system/equipment operates correctly and meets safety requirements when subjected to external influences (e.g., climatic conditions, EMI/ESD, power supply variations)

- Safety-related application conditions — describes the rules, conditions and constraints to be observed (e.g., precautions in installation/testing, safety warnings)

- Qualification Testing.

3.9.2.4 Safety Integrity Levels - Safety integrity relates to the likelihood of the system/equipment meeting its safety requirements. Guidance is provided on the relationship between safety integrity levels and the quality management, safety management, and technical safety activities. The safety integrity level is actually to be assigned as part of the safety management activity via a hazard analysis and risk assessment.

3.9.2.5 System Acceptance - An independent safety assessment is to be carried out on the system/equipment to provide additional assurance of safety. This may involve the conduct of additional safety (verification/validation) activities. Guidance as to the depth of this assessment and the interaction with the manufacturer is to be found in the CENELEC document TC9X-WG5B, "Dependability for Guided Transport Systems, Part 4: Specification and Demonstration of Safety."

The Railway Authority will then base their acceptance of the system/equipment on the proof-of-safety or safety case plus the results of the independent safety assessment.

3.10 ABB SIGNAL AB

ABB Signal AB. located in Stockholm, Sweden, is one of the first companies in the world to develop computer-based systems for safety critical railway applications. ABB Signal works in close cooperation with one of their major customers, the Swedish State Railways, in the development of computer-based systems including automatic train control and signalling (e.g., interlocking) products. Currently, the design philosophy of ABB Signal's safety critical computer-based products is based upon the use of a single channel computer structure with diversified software that is developed by separate design teams.

ABB Signal and its Fail-Safe Department have no role in the approval of railway equipment used in Sweden (or elsewhere). Their internal safety review activities are directed towards their own products, which in turn are subject to third-party assessment and customer (railway authority) approval. In Sweden, the Swedish national railway authority participates in safety reviews and analyses of the products and gives final approval for use. This participation is usually on the system level (e.g., review of Functional Requirement Specifications), but may also be more detailed in some instances.

There are no national standards in Sweden (such as in Germany, France or England) for the design/assessment of computer-based railway equipment/systems. There are, of course, signalling rules for governing train movement. ABB Signal has developed and has been using their own internal standards and guidelines for computer system design and safety assurance. One key internal ABB Signal document for safety critical systems is entitled "Design of Fail-Safe Equipment: Organization of Safety Measures in Different Product Phases" (FS 2059). This document contains proprietary business information, and thus cannot be addressed in detail here. However, in general, it describes the safety-related activities that should be conducted during the two main phases of a product – development (including manufacturing) and use (including modifications and repairs).

3.10.1 The Safety Review (V&V) Process

The Fail-Safe Department of ABB Signal is separate from the development group, and has primary responsibility for the safety review process which includes safety verification and validation activities. The key activities in the safety review process, which is integrated into the overall development process, include the conduct of safety reviews, the conduct of safety analyses, the preparation of a Safety Description, and the conduct of safety audits. It should be emphasized that the safety review process described here does not cover all safety activities performed on a product/system; one such example is testing. This is addressed in more detail in the proprietary document referenced earlier.

3.10.1.1 Safety Reviews - All documentation pertinent to safety is subjected to a safety review. This includes the following items:

- (Customer) Requirement Specification
- Functional Requirement Specification
- Functional (System) Description
- (Program) Block Specifications (S/W)
- Device Specifications (H/W)
- Electrical and mechanical design documentation (H/W)

- Safety analyses
- (System) Test Specification
- System Test Record
- (Manufacturing) Test Instructions (H/W)
- Safety Description
- Plant data
- Plant test data
- Plant documentation
- Plant test procedures
- Plant Test Record.

The purpose of these reviews is as follows:

- To ensure that the safety requirements are defined and met at all stages in the development process
- To uncover safety problems as early as possible
- To bring the safety of the product under scrutiny of as many qualified and experienced persons as possible
- To ensure that applicable safety principles, standards, rules and procedures are stipulated and adhered to
- To ensure that the document under review is correct and complete with regard to safety
- In some cases, to gain customer approval of the document under review with regard to safety.

These safety reviews are normally performed separately from other activities (e.g., design reviews), and the results of the reviews are documented in meeting minutes and archived for later reference.

In addition, code reviews are employed only for non-diversified software to help ensure correctness. As mentioned earlier, ABB's philosophy is (typically) based upon software diversity, and code reviews are not conducted for this software structure.

Safety reviews are performed throughout the design process by a safety review committee that is composed of design engineers, safety specialists including Fail-Safe Department personnel, and others as defined in a project organization plan. Railway traffic and signalling experts participate in the review of the Requirement Specification. Customer representatives participate in the Functional Requirement Specification review.

3.10.1.2 Safety Analyses - Safety analyses are performed on the electrical design to ensure that safety requirements are met and to provide a proof-of-safety for the device in question. Techniques used include a failure modes and effects analysis and/or fault tree analysis. Such analyses are performed in the implementation phase, after prototypes are ready for verification. It is recommended that these analyses are initiated as early as possible and updated as the design progresses. The engineers who are responsible for the Device Specification typically perform these analyses.

3.10.1.3 Safety Description - The Safety Description, prepared by the engineer in charge of project safety, describes the principles employed in the design to achieve safety. On the system level, a fault tree is typically used to show functional and system safety achieved in software and hardware and the protective measures taken. This description is prepared early in the specification phase, and is finalized at the completion of the implementation phase.

3.10.1.4 Safety Audits - Safety audits are performed exclusively by Fail-Safe Department staff at various times in the product life cycle. The exact timing depends upon the length and scope of the project. At a minimum, a safety audit is held as part of the final assessment, prior to delivery of the product to the customer. The purpose is to ensure adherence to routines and procedures established for the management of projects and departments involved in the development and production of fail-safe products/systems, and to improve these routines and procedures.

3.11 SIEMENS AG

The Siemens AG transportation division, located in Braunschweig, Germany, has responsibility for the development of Siemens' safety critical signalling and train control systems including computer-based systems/equipment. Siemens' signalling/train control equipment is used extensively by the German Federal Railway (DB) on their conventional and high-speed Intercity Express (ICE) lines as well as in other worldwide locations. It is also being used in the Transrapid maglev system which is under test at the Emsland test facility in Germany.

The basic design of Siemens' safety critical computer-based systems (e.g., interlockings, train control) is based on the SIMIS microcomputer architecture, which utilizes either a dual or triple channel computer configuration to help ensure safety of operations.

The approval process in Germany for Siemens' safety critical railway systems is the same as those processes described in the TÜV Rheinland and German Federal Railway sections of this report. For equipment supplied to the DB (the primary customer of Siemens), the Bundesbahn Zentralamt (BZA) office in Munich assesses the safety evidence provided by Siemens and conducts additional testing as necessary before granting approval for use of the equipment. For other applications, an independent assessor reviews the safety evidence and recommends approval to a local government body, who then grants final approval for use of the equipment. In both instances, agreement is obtained up front on the development and safety assessment (verification and validation) processes to be utilized.

The primary standards/guidelines utilized by Siemens in the development and assessment of safety critical computer-based systems for the railway industry are as follows:

- DIN V VDE 0801, "Principles for Computers in Safety Related Systems"
- DIN V 19250, "Fundamental Safety Analyses for MSR (Measurement-Control-Regulation) Protection Devices"
- Mü 8004, "Principles for Technical Approval for Signalling and Communications Technology"
- A25000-P0001-A001-01-0035, "Software Development Guidelines: Software for Computers in Safety Critical Applications"
- "Chapter 3, Assessment Methods for Safety Critical Software by Siemens AG."

The first two documents (i.e. DIN V VDE 0801 and DIN V 19250) are addressed in more detail in the section of this report on TÜV Rheinland, and the third (i.e. Mü 8004) in the section on the German Federal Railway. When supplying equipment to the DB, Siemens focuses heavily on Mü 8004.

The latter two documents cited above are part of Siemens' internal standards. The document A25000-P0001-A001-01-0035 (38 pages) is Siemens' standard/guideline for the development of software in safety critical applications. It addresses a development methodology and associated tools. A copy of the document was received in the German language, but it is proprietary in nature; thus, it is not addressed in more detail here. The other internal guideline entitled "Chapter 3, Assessment Methods for Safety Critical Software" is just what the title implies -- a description of methods to assess and ensure the safety of software in safety critical applications. According to Siemens staff, it represents an enhancement of the procedures in Mü 8004. It was received in the German language, and the English translation was not available in time for an overview to be generated for this report.

According to Siemens staff, the process used to assess the safety of the overall system and hardware is identical to that described in Mü 8004.

3.12 MATRA TRANSPORT

Matra Transport, with headquarters in Montrouge (suburb of Paris), France, designs automatic train control (ATC) systems for both attended and unattended (driverless) urban transport applications. One of Matra's driverless, computer-based ATC systems is the VAL system, in operation since 1989 in Lille, France. This technology is also being used in numerous other locations (e.g., Jacksonville, Fla., Orly Airport, Chicago O'Hare Airport, Taipei). Since that time, other computer-based ATC systems have been developed including SACEM (Line A of RER in Paris, Lines A and 8 in Mexico) and MAGGALY (Line D of Lyon's heavy rail system). Two other ATC systems under development include METEOR (planned for the Paris Metro) and ANTARES or KVIM/KVBP (a SACEM-like system planned for Line C of the RER -- operated by SNCF).

For transit projects in France, a Safety Committee is formed to review and approve the design with respect to safety. These committees are typically composed of the client representatives, a government agency, and possibly independent university and/or consultant members. In the case of SACEM, the RATP (Paris Metro) "defended" the safety of the system. In other cases such as VAL (in Lille) and MAGGALY (in Lyon), Matra played a key role in "defending" the safety.

In the development of the SACEM ATC system, a significant effort was performed relative to validating the software. This effort, which involved the use of formal methods/proofs, represents one of the more sophisticated software validation processes (used in the transportation industry) identified during this study. A brief overview of this process is provided later in this section.

Matra's computer-based ATC systems are currently based upon a "coded monoprocessor" technique. In this technique, which is based upon a single channel microprocessor concept, three main functions are performed: coding of fail-safe inputs, coded data processing, and output decoding. All data are encoded, resulting in coded variables that have both a functional and coded part (or signature). The signatures are initialized at the beginning of the program. Each elementary operation is replaced by a coded operation such that the signature of the results can be predetermined. A special device referred to as a Dynamic Controller uses signatures stored in a PROM to verify the correctness of computer operations. Vital outputs receive power only if processed data have the correct signatures. Matra is also developing a second generation coded monoprocessor system known as "Transputer."

A safety plan is prepared by Matra for each project. These plans outline the activities to be performed within Matra to ensure the safety of the system/equipment being developed, and incorporate safety requirements such as those imposed by applicable standards. Applicable French standards in the area of safety are addressed in the section pertaining to the SNCF.

Certain safety verification and validation activities are performed by Matra's Hardware and Software Development Groups. In addition, the RAMSS (Reliability, Availability, Maintainability, Safety and Security) Division works with the development divisions to ensure the systems meet requirements in these areas. Activities are performed throughout the development cycle.

3.12.1 Safety Verification and Validation Process

Safety activities performed by Matra can be separated into three primary areas: system/subsystem safety activities, hardware safety activities, and software safety activities.

3.12.1.1 System/Subsystem Activities - The following six major safety activities are typically performed at the system/subsystem level:

- Preliminary Hazard Analysis (PHA)
- System Hazard Analysis (SHA)
- Operating Hazard Analysis (OHA)
- Identification of system safety functions
- Functional Safety Analysis, and
- Dimensionment Analysis.

3.12.1.1.1 Preliminary Hazard Analysis - A PHA is performed to identify system hazards, taking into account the potential loss of certain functions, and to propose solutions to eliminate, reduce, or control the risks. Starting with final consequences, scenarios leading to those consequences are identified. Using the System Specifications, elements that can lead to the system hazards are then listed. Actions and safety criteria which must be taken into account in the design or in the operating/maintenance procedures are defined.

3.12.1.1.2 System Hazard Analysis - An SHA is performed to determine potential safety problems of the total system design, including human errors. A system model is constructed to illustrate the high level architecture of the system, the breakdown into subsystems and their relationships, and human actions. All possible functional consequences of the different hazardous events are examined. Then, actions and safety criteria are proposed to eliminate, reduce, or control the identified risks.

3.12.1.1.3 Operating Hazard Analysis - An OHA is conducted to identify operating-related hazards and recommend risk reduction actions. First, all documents describing the intended use of the system (e.g., operating/maintenance instructions) are reviewed, and the potential impact on safety is determined for each instruction. The results of the PHA and SHA facilitate this process. Modifications are proposed as necessary.

3.12.1.1.4 Identification of System Safety Functions - All system safety functions in the System Specification are examined in order to determine those which relate to hazards identified in the PHA and SHA. The result is a Safety Functions List and a list of the safety criteria for each function.

3.12.1.1.5 Functional Safety Analysis - This analysis involves the generation of detailed functional safety criteria which must be considered in the design and implementation of the hardware and software and development of the operating/maintenance instructions. These criteria serve as input for further hardware/software safety analyses and for the Dimensionment Analysis.

3.12.1.1.6 Dimensionment Analysis - This analysis verifies that the parameters which determine system performance (e.g., deceleration rate for emergency braking) are compatible with the safety requirements and with external restraints (e.g., train characteristics). Mathematical models are used to verify that such actions as train speed and stopping points are respected, even in worst case situations. Constants/variables that describe the system configuration/functioning are verified to ensure they meet performance criteria established by the Functional Hazard Analysis. Specific constraints such as train speed are verified for each switching point. Also, the coherence between the railway map and constants file (that describes it) is verified.

3.12.1.2 Hardware Safety Activities - Each safety critical hardware unit is analyzed to ensure that any failures or combination of failures lead to a safe state. There are two major activities performed here: Hardware Board Analysis and Hardware Interface Analysis.

3.12.1.2.1 Hardware Board Analysis - This analysis is conducted to ensure that the operation of the hardware boards do not violate the safety criteria. Effects of failures are observed by inserting a failure on a prototype. A simulation technique is used in more complex systems. This is often accompanied by a failure modes, effects and criticality analysis (FMECA) and a fault tree analysis (FTA).

3.12.1.2.2 Hardware Interface Analysis - This analysis is performed to ensure that the hardware boards are used correctly in the entire system. This is done by cross-reading the specification documents (those which specify the intended use of each hardware unit in the system) with the specifications of the hardware unit itself.

3.12.1.3 Software Safety Activities - The software safety approach used by Matra is heavily based upon the coded monoprocessor concept described earlier. This concept is intended to cover all kinds of errors (except perhaps certain types of design errors) ranging from compilation problems to real time errors in instruction execution. To ensure that the software is as error-free as possible, Matra uses a combination of formal software development techniques along with software analyses/testing.

The formal techniques are applied in two main steps. First, a formal (mathematical) description is prepared of the required properties of the software, followed by a computer-aided-proof of these properties. Manual proofs are required in some instances.

The following eight activities are typically used by Matra, in addition to formal methods, to help ensure software safety:

- Identification of safety software items
- Software FMECA
- Search and evaluation of non-safety scenarios
- Coded software analysis
- Safety constants validation
- Critical reading of applicative code
- Safety software units testing, and
- Safety software items functional testing.

3.12.1.3.1 Identification of Safety Software Items - Using safety criteria from the Functional Safety Analysis, safety critical software items are identified and specific software criteria are established for those items.

3.12.1.3.2 Software FMECA - Using an approach similar to an FMECA, an analysis is performed on the non-coded software to identify software "failures" (errors in the normal execution of the program) and undetected errors. Errors in the specially coded software are not addressed by this analysis as they are covered by the signature check. All error types are examined in an inductive manner with the help of software analysis tools (e.g., flow charts). The seriousness of the consequences of the errors and their detectability are determined.

3.12.1.3.3 Search and Evaluation of Non-Safety Scenario - Using a list of undetected errors from the software FMECA, a search is conducted to identify possible unsafe effects of multiple errors and to determine their probability.

3.12.1.3.4 Coded Software Analysis - This analysis is performed to verify the correct usage of certain "state-of-the-art" implementation rules for the code. Aspects of interest include the type of safety constants and variables used, the use of structured data, addressing of constants, and vital message processing.

3.12.1.3.5 Safety Constant Validation - This effort involves verifying that the constants actually used by the software reflect those that describe the relevant system parameters. The latter were validated by the Dimensionment Analysis.

3.12.1.3.6 Critical Reading of Applicative Code - This task involves reading and examining the source code to determine that the various safety criteria are correctly taken into account.

3.12.1.3.7 Safety Software Units Testing - This task involves two steps, and is conducted to ensure that the different software units are consistent with their design documents and safety criteria. Step one is based on a functional or "black box" approach. A set of scenarios is prepared (including specific inputs and expected outputs) using the preliminary design document, and different software unit functions are tested accordingly. Both "average" and out-of-bound situations are addressed. Step two is based on a structural or "white box" approach with the help of specific software tools.

3.12.1.3.8 Safety Software Items Functional Testing - The purpose of this task is to ensure that the software items meet all safety criteria in all circumstances. Test scenarios are prepared based upon the safety criteria and functional specification. All safety critical branching possibilities of the software item must be covered in "average" and boundary situations as well as in certain out-of-bound conditions. The latter is conducted to verify the robustness of the software with respect to the safety criteria.

3.12.2 <u>SACEM Software Validation</u>

As mentioned earlier, a significant effort was undertaken by Matra to apply formal software validation techniques to the SACEM computer-based ATC system in use on the RER Line A in Paris. In SACEM, each train receives vital information (e.g., speed, position, obstacles, switch status, distance-to-go and speed limit) from wayside equipment, computes its location in the network and the maximum safe speed (via tachometers and beacons), and protects against overspeed situations by commanding emergency braking. Two other manufacturers (i.e., GEC-Alsthom and CSEE Transport) plus the RATP (Paris Metro) and SNCF (French National Railways) assisted Matra in the overall development of SACEM.

The software in SACEM was written in accordance with the coded monoprocessor concept described earlier. In this application (RER Line A), SACEM has approximately 21,000 lines of code written in the Modula 2 language. The intent of the validation process was to make the software as error-free as possible. The validation process was based on four main principles which are summarized on the next page.

3.12.2.1 Inspection - Four different teams from the manufacturers' organizations were involved including the design team, a safety team, the validation team and a "formal re-expression" team. In addition, the RATP formed their own independent teams. These teams performed various reviews and inspections throughout the validation effort.

3.12.2.2 Semi-Global and Global Tests - Both semi-global and global tests were performed. The former was performed to reproduce procedural behavior of the software, while the latter (based upon real operating scenarios) utilized a special simulator to perform real-time checks of system functions.

3.12.2.3 Formal Proof - A formal proof was performed which involved showing (by hand) that every procedure carried out a process which was correct with respect to the specifications.

3.12.3.4 Formal Re-expression - A formal specification or "re-expression" was written based upon errors detected via the formal proof.

3.13 SASIB

Sasib Societa Per Azioni (Spa), located in Bologna, Italy, supplies conventional and computer-based signalling/train control and other equipment/systems for the railway industry. One of their biggest customers is the Italian Railways (FS). Sasib is the parent organization of General Railway Signal (GRS) in the United States.

In general, manufacturers (or contractors) in Italy must demonstrate (e.g., via testing) that all specifications including those pertaining to safety have been met, and must provide associated documentation that the system has a "fail-safe" behavior under ali circumstances. The Italian Railways typically verifies the results of the tests, and carries out additional tests or checks as deemed necessary.

SASIB currently has their own verification and validation methodology that was applied during the development of their first generation computer-based interlocking system (also known as the ASCV project). The methodology is comprised of a number of analysis and test activities that are highly integrated into the development process from system requirements through field testing. A brief overview of this methodology is provided later in this section. During the development of this interlocking system, SASIB's client (i.e., Italian Railways) had different teams of specialists (in hardware. software and system aspects) to continuously monitor the development and conduct various review/analysis/test activities as desired.

SASIB has been reviewing their process and are making some revisions due to three main factors: 1) attention being directed to ISO 9000 quality standards, 2) the new CENELEC safety standards being developed by the European Community, and 3) inputs/desires of their main customer – the Italian Railways. In particular, SASIB is developing a new methodology for trial application in the development of safety critical software for their updated ASCV interlocking system with hot stand-by capabilities (known as the S. R. ASCV project). Should the concept show viability, SASIB intends to refine it and use it as a company-wide Software Quality Management System (SQMS).

This new concept (described in the SASIB document SRAS-02-S-000-3) is based on the content of over 14 existing U.S., European and international quality, software development, and safety standards such as ISO 9001/EN29001/BS 5750; IEC 65A 122; RIA Tech. Spec. No. 23; ANSI/IEEE Std 730, 828, 829 and 830; IEEE Std. 1008, 1012, 1016-1987 and 1028; BS 6719:1986; and ESA-PSS-05-0 Issue 1. In general, the SQMS approach involves the definition of a set of activities pertaining to the overall project and the entire software life cycle. Activities include those pertaining to software design, configuration management, quality control, progress monitoring, testing and others. A software verification and validation plan (prepared by a third party organization) is to be a part of this effort. Recommended software V&V activities and related tools and techniques for the new S. R. ASCV interlocking project are described in the SASIB document SRAS-03-S-000-4. Because of the experimental nature of the software V&V plan, a third party organization is to cooperate with SASIB's staff in the preparation and implementation of the plan. The customer will provide consultive technical information throughout the project, and will be kept informed as to phase completions and overall project progress.

SASIB's future plans are to extend the SQMS to a Safety Software Management System (SSQMS) in which safety aspects can be designed into a product and traced, in an integrated manner, as defined in the new CENELEC standards. In this approach, a customer (e.g., Italian Railways) would verify project progress and would have the following responsibilities:

- Review of system requirements
- Early definition of acceptance criteria
- Random auditing of different parts of the system and project documentation
- Walkthrough of the safety aspects and architectural choices
- Verification of results and execution of acceptance tests
- Review of random tests regarding ORE A-155 standards, and
- Field testing of suitability.

On a further note, the Italian Railways has produced the document IS 402, "Technical Specification for the Supply of Electronic Equipment for Safety and Signalling Systems." This document pertains to electronic railway equipment in general and addresses such topics as accepted international standards, general rules to be followed by manufacturers, documentation, testing, packaging rules and others. It generally does not address safety verification or validation, and, therefore, was not described in any further detail here.

3.13.1 SASIB'S Current V&V Methodology

The verification and validation methodology as utilized by SASIB and the Italian Railways in the development of the ASCV computer-based interlocking system consists of three major stages: prototype realization, prototype test, and prototype installation. Associated activities performed by both SASIB and the Italian Railways (FS) in this development effort are described below.

3.13.1.1 Prototype Realization - Major phases and activities performed in the prototype realization stage of development were as follows:

- System Requirements Phase

  SASIB – Prepared system requirements specification.

  FS – Provided feedback and approval to SASIB.

- System Design Phase

  SASIB – Prepared architectural and functional definitions of the system. Prepared software functional description based upon the DeMarco structured analysis method.

  FS – Reviewed and verified compliance with system requirements. Provided feedback and approval to SASIB.

- Software Design Phase

  SASIB – Prepared software design description. Every function was detailed by describing all logic, variables, and data involved. Special attention was given to technical solutions involving safety aspects. Prepared a flow chart of the logic.

  FS – Reviewed and verified compliance with the functional description. Performed critical analysis of the safety and fault-tolerant solutions. Provided feedback and approval to SASIB.

- Coding and Testing Phase

  SASIB – Coded, designed and debugged system. Utilized two different software testing methods (i.e., black box and white box). In the black box method, functions defined by the software design were tested. In the white box method, code was tested from a structural point of view. Test coverage was 100% in that all functions and branches were tested. Positive and negative testing techniques were performed in which outputs were shown to be correct or incorrect as a result of correct or incorrect inputs, respectively. Testing was conducted by an separate team, independent from the development staff.

FS – Conducted static walk-throughs of entire code, focusing on safety related functions. Performed dynamic walk-throughs on selected portions using emulation tools. Conducted FMEA by defining and considering all possible hardware failures and their effects upon software, and determining if failures were detected or could be tolerated. Provided feedback and approval to SASIB.

- Hardware Design Phase

  SASIB – Designed hardware and conducted functional tests on vital and nonvital circuits.

  FS – Conducted safety analysis of each fail-safe circuit in the laboratory according to ORE A-155 standards (i.e., in accordance with ORE Report A-155.2/RP 9). Performed functional performance evaluations on arbitrary sample. Provided feedback and approval to SASIB.

- Integration Phase

  SASIB – Established performance measures and conducted integration tests.

  FS – Verified the logical correctness of the system.

3.13.1.2 Prototype Test - This stage involved a test phase in which the system was tested by SASIB to verify its compliance with applicable standards. FS conducted testing in accordance with IS-402 to verify the "robustness" of the system with respect to vibrations, EMC, EMI, overvoltage and overcurrent.

3.13.1.3 Prototype Installation - In this stage of development, the system was tested in the field in conjunction with a conventional electromechanical system. Tests were conducted using an exerciser (to emulate system inputs) and data logger (to record and compare behavior of two systems). Major phases and activities performed during prototype installation were as follows:

- Exerciser and Data Logger Requirements Specification Phase

  SASIB – Defined requirements for the exerciser, data logger and field test execution.

  FS – Provided feedback and approval to SASIB.

- System Design Specification Phase

  SASIB – Prepared exerciser and data-logger functional specification.

  FS – Did not participate.

- Design Phase

  SASIB – Performed exerciser and data-logger design.

  FS – Did not participate.

- Installation Phase

  SASIB/FS – Outputs were checked under various input conditions.

3.14 SNCF

The SNCF (French National Railway), with Headquarters in Paris, France, utilizes computer-based and conventional signalling and train control systems to help ensure the safe operation of their trains, including the TGV.

SNCF plays a key role in the process of certifying/approving new systems/equipment for use on its railway. Typically, this involves working closely with the manufacturer throughout the development process as well as implementing a certification process, which includes extensive testing. The certification file (results) must be submitted to the Ministry of Transport, who then gives final approval for use in revenue service.

Although provisions were put in place to receive detailed information concerning SNCF's activities/standards in the area of safety verifications and validations, such information was not available for this report. However, it is known that there are at least three French standards (published by the Railway Standards Bureau) that pertain to computer-based systems and software. Those are as follows:

- NF F 71-011, "Software Dependability-General Information"
- NF F 71-012, "Software Dependability-Stresses on Software"
- NF F 71-013, "Software Dependability-Adapted Methods for Software Safety Analysis."

English versions of these standards were not available for this interim report, but their general nature is known from other documentation. NF F 71-011 provides general information pertaining to such topics as classification of the safety system, terminology, interactions/constraints between hardware and software, and software project organization. NF F 71-012 addresses such topics as formal specifications, risk assessments, software analyses, and documentation. NF F 71-013 deals with software reviews and inspections. In addition, it is known that one of the key internal standards within SNCF is DEI 100 (this document was also not available for this study).

One document that was available for this report describes the general process utilized in the validation and certification of the computer-based TVM 430 train control system for the TGV NORD high-speed line. This process is summarized below to serve as an example of the safety activities (including verification and validation) performed by both the manufacturer and, especially, SNCF.

3.14.1 Validation and Certification of the TVM 430 Control System

TVM 430, developed by CSEE-Transport (also of France), is a computer-based track-to-train signal transmission system that implements cab-signalling and speed-regulating functions. It essentially allows for the on-board calculation of maximum speed based upon current train location as well as speed and target distance information continually received from the wayside. The system is structured around a dual redundant computer configuration with a software "voting" or comparison computer, based upon the "coded monoprocessor" technology utilized in the SACEM train control system (used on Line A of the RER in Paris).

3.14.1.1 Validation/Certification Process - Validation was conducted by the manufacturer, and involved demonstrating that the system complied with (especially) safety specifications provided by SNCF. Certification was the responsibility of SNCF, and involved monitoring the supplier's development process, reviewing all safety results from the different development phases, and conducting additional testing as required to fully demonstrate safety. Further, an independent organization was secured to perform a safety audit of the supplier's organization and safety documentation. As mentioned above, the certification file was provided to the Ministry of Transport.

The TVM 430 system was developed by CSEE-Transport under a quality assurance plan (or PAQ) that is recommended in the French AFNOR Z67 130 Standard, "Recommendation for a Software Quality Plan." This process involved the use of the "V" development cycle in which the system is broken down into subsystems and smaller software and hardware components (one side of the "V"), and then tests are periodically conducted to verify compliance with specifications (other side of "V"). SNCF personnel reviewed and approved the documentation/results of each phase of the development.

In addition, the manufacturer submitted "safety document files" to SNCF for review. These files describe key aspects of specific technologies/design techniques utilized to help ensure safety. One example is that of the coded monoprocessor technique.

3.14.1.2 Validation Testing - Although details of the validation activities performed by the manufacturer are not available at this time, it appears that the manufacturer must conduct (pass) a quality test (demonstrate adherence to the PAQ), and must conduct safety verifications. The latter includes verification of the functional specifications and verification that principles directed to ensuring system safety are being followed.

3.14.1.3 Certification Testing - In addition to monitoring the system development, four types of certification testing was performed by SNCF:

- Hardware testing
- Generic software testing
- Total system testing
- Application specific testing.

3.14.1.3.1 Hardware Testing - A mockup/prototype of the system hardware was tested at SNCF's facilities to ensure compliance with specifications. Further, the fail-safe portions of the system were tested via a mockup in order to examine the effects of failures. Test reports were generated for each card.

3.14.1.3.2 Generic Software Testing - The generic algorithm of the TVM 430 system was tested by two means. First, the software was tested via a special tool to identify certain features (e.g., testability, complexity) and to compare the software against its design documents. Second, the software was tested (validated) via a simulator to determine compliance with specifications. These tests were done in addition to those conducted by the manufacturer.

3.14.1.3.3 Total System Testing - SNCF organized and conducted tests on the overall system at a special test site. The purpose was to ensure correct operation of all system functions. In addition, a failure modes, effects, and criticality analysis was performed on the most critical portions of the system.

3.14.1.3.4 Site-Specific Testing - On-site and bench tests were performed on the actual site specific software, the latter using an environment simulator.

In summary, SNCF's certification process is based upon strict adherence to a design process, in-depth theoretical analysis, as well as laboratory, bench, and on-site testing.

3.15 MINISTRY OF DEFENCE

The Ministry of Defence (MOD), within the United Kingdom, has issued the following two standards (in three parts) pertaining to the safety of programmable electronic (computer) systems in defence applications:

- Interim Defence Standard 00-55, "The Procurement of Safety Critical Software in Defence Equipment, Part 1: Requirements," Issue 1, April 5, 1991
- Interim Defence Standard 00-55, "The Procurements of Safety Critical Software in Defence Equipment, Part 2: Guidance," Issue 1, April 5, 1991, and
- Interim Defence Standard 00-56, "Hazard Analysis and Safety Classification of the Computer and Programmable Electronic System Elements of Defence Equipment," Issue 1, April 5, 1991.

The first (i.e., 00-55) deals with software while the latter (i.e., 00-56) is more systems-oriented. Overviews of these two standards are provided below.

3.15.1 <u>Interim Defence Standard 00-55</u>

This interim standard describes procedures and requirements for the development and assessment of safety critical software – that software used to implement safety critical functions. It is intended to apply to the specification, design, coding, production and in-service maintenance/modifications of the software. It defines a software development process in which V&V activities are an integral part, and involves the use of formal methods with dynamic testing and static path analysis. The standard is actually in two parts. Part 1 defines the general software requirements while Part 2 provides additional guidance for meeting those requirements.

Part 1 is divided into two main sections: safety management and software engineering practices.

3.15.1.1 Safety Management - A number of key safety management requirements relating to the assessment/V&V of software are described. One is the need for a hazard analysis and risk assessment to be conducted as soon as possible in the procurement process to, among other things, identify the safety critical portions of the software. This analysis/assessment is to be conducted in accordance with that described in Interim Defence Standard 00-56. Another key requirement is that a V&V team is to be formed independent from the design team, and is to prepare (and implement) a Verification and Validation Plan. There is also to be an Independent Safety Auditor (independent from the design team) who is to review all documentation and to ensure compliance with this and any other applicable standards/guidelines/codes. The Independent Safety Auditor is also to endorse the Safety Critical Software Certificate (as produced by the design team) which is provided to the MOD for acceptance/approval.

3.15.1.2 Software Engineering Practices - As described earlier, verification and validation activities are a key part of the software engineering process. Examples of the V&V activities performed by the design and/or V&V teams are described below:

- Specification – formal methods are to be used to generate a Formal (Software) Specification, which is to be subjected to a verification; the Formal Specification is checked for syntactic and type errors by using suitable tools; a preliminary validation is also to be performed in which the Software Specification is checked for compliance with the Requirements Specification; Formal Arguments are checked by the V&V team

- Design – the Formal Design, again based upon formal methods, is to be verified for syntactic and type errors using suitable tools; Formal Arguments and Proof Obligations are to be checked by the V&V team

- Coding – a verification is to be performed using a Static Path Analysis on the source code, and is to include a control flow analysis, data use analysis, information flow analysis and, where appropriate, analysis of program constructs; dynamic testing is to be conducted on the code

- Formal Arguments – to include a verification/review of Formal Arguments and Formal Proofs

- Dynamic Testing – to include dynamic testing of all individual modules, partly integrated groups and the entire integrated safety critical software product

- Validation – validation testing is to be conducted to determine compliance with the Software Requirements Specification; test results are to be checked for correct functionality, correct actions on errors, correct timing and conformance to other non-functional requirements.

3.15.2 Interim Defence Standard 00-56

This interim standard describes the techniques and procedures for conducting a hazard analysis and risk assessment on new systems and following modifications/maintenance on existing systems. The primary purpose of this analysis/assessment is to identify and evaluate hazards within the system in order to determine the maximum tolerable risks. Methods for reducing risks are described in other documents. Further, verification and validation of the system is not addressed by this standard.

Hazard analysis and risk assessment activities are to be carried out by the design team. There is also to be an independent safety auditor, who is to audit the entire project, ensure compliance with this standard and conduct an independent safety assessment on selected features of the system.

The concepts/relationships of accident severity categories, risk classifications, risk classes and safety integrity levels are discussed, and hazard analysis/risk assessment activities are defined for various risk classes/safety integrity levels.

The major hazard analysis and risk assessment activities to be conducted are as follows:

- Preliminary Hazard Analysis
- System Hazard Analysis
- Functional Hazard Analysis
- Zonal Analysis
- Component Failure Analysis
- Operating and Support Hazard Analysis
- Occupational Health Hazard Analysis
- System Risk Analysis
- Failure Probability Analysis (fault tree analysis)
- System Change Hazard Analysis.

Guidance for the use of these techniques in different life cycle phases and resulting documentation is provided in appendices to the standard.

3.16 INSTITUTE OF RAILWAY SIGNAL ENGINEERS

In 1990, the Institution of Railway Signal Engineers (IRSE) established a Technical Committee (TC) of senior engineers from railway authorities and suppliers/contractors in eight different European countries (i.e., Austria, Belgium, France, Germany, Italy, Netherlands, Switzerland, and the U.K.). The task was to study the possibility of identifying a common standard for a proof of safety for new technology (computer-based) signalling systems. More specifically, the three major subjects of interest were 1) a definition of a proof of safety, 2) the identification of common standards, and 3) the identification of methods and procedures to be used for overall system assurance. The work included the identification and review of existing safety standards in Europe for railway applications, and resulted in a number of recommendations which are documented in the report entitled "Report No. 1, Safety System Validation with Regard to Cross Acceptance of Signalling Systems by the Railways," dated January 14, 1992.

The following general safety assurance requirements of a signalling system were identified:

- Total system safety must be considered
- Degree of safety required must be determined
- Minimum standards are needed for each degree of safety (of which five levels were defined by the committee)
- Total life cycle of the system must be considered
- Strict management is needed of all processes involved from requirements through post-development modifications.

The following set of proposed rules go along with the above requirements:

- A new system must be at least as safe as the one it replaces
- A formal and well documented process is needed throughout the life cycle
- Safe operation is needed over the entire range of environmental conditions expected
- A single failure must not result in an unsafe condition
- Any latent failure that could react in combination with a second failure to produce an unsafe condition must be detected, and a safe response ensured.

3.16.1 Proof of Safety

An overall process was approved by the committee for producing/approving a proof of safety for the system/equipment. The contractor is to provide the proof of safety which is to consist of two parts. One part is to be a verification, in which the products of each phase of the development cycle are shown to comply with the requirements of that phase. The other part is to be a validation, in which the complete system is shown to comply with requirements. International standards are to be utilized, and certain procedures are mandatory to establish the proof of safety for specific assigned degrees of safety. Finally, the railway authority is to be responsible for endorsing the proof of safety via an audit or other means.

All processes to be carried out during the system life cycle are to be documented in accordance with a quality assurance standard such as BS 5750 or EN 29001, and the full set of documentation comprises the proof of safety.

The proof of safety process consists of the following major activities:

- Preliminary activities – safety functions of the system, hardware and software are determined; specific assessment methods/activities are selected; integrity levels are assigned; system, hardware and software architectures are selected; safety and quality plans are developed

- Requirements Specification – a requirements specification is developed

- Verification – verification team prepares and implements a verification plan in order to determine that the products of each phase complies with requirements for the phase relative to correctness, completeness, consistency and accuracy

- Validation – determine that the completed system complies with requirements in accordance with a validation plan

- Assessment – independent party performs an audit to ensure that safety plan has been followed

- Documentation Review – documentation which should include such items as requirements and functional specifications, hazard analyses, system descriptions, acceptance test specifications and O&M manuals are reviewed.

Tables are provided which separately identify and classify various types of verification/validation methods for hardware, software and the system. Three classifications are used: mandatory, highly recommended and recommended. A wide variety of methods are identified including FMECAs, fault tree analysis, white box testing, software errors effects analysis, static and dynamic analysis, hazard analysis and many others. An appendix to the document provides additional information on the nature and application of the methods.

A number of specific recommendations are provided by the IRSE committee. Just a few key recommendations are listed below:

- Available international standards such as IEC 65A should be followed by the railways and contractors

- Configuration and design of the system is to be aided by the use of CASE tools

- Hardware and software is to be validated, but software need not be validated if sufficient diversity is present

- Consideration is to be given to the use of formal methods including a mathematical proof of safety.

3.17 RAILWAY TECHNICAL RESEARCH INSTITUTE

Several different types of safety critical computer controlled signalling systems are used by the Japan Railways. This includes a solid-state interlocking, an automatic train control (ATC) system, an automatic train stop system with a transponder, and a level crossing alarm controller. The Railway Technical Research Institute (RTRI) in Tokyo, Japan, has been involved in both the design and assessment (safety verification and validation) of many of these signalling systems. Safety verification and validation as performed by the RTRI is typically done at the request of the user railway or manufacturer.

The signalling system development process is different from that used in the United States. In Japan, an organization such as the RTRI designs/develops a product based upon a request from the Railway, and then provides the design/drawings to a manufact'irer who then builds the equipment. This is different from the process used in the U.S. where manufacturers design and build their own products and produce their own drawings based upon (functional) specifications from a user.

3.17.1 Safety V&V Process

The safety management (including safety V&V) process used by the RTRI is actually a modified version of the process/activities identified in MIL-STD-882. This modified process (summarized below) is documented in an internal system safety management manual. This manual is used in conjunction with a guideline on fail-safe and fault tolerant technology to ensure the safety of computer-based systems. The design guidelines comprise 96 requirements that pertain to all phases of a system's development. An example of the design and safety assessment of a computer-based signalling system (i.e., SMILE interlocking system) is provided later in this section. As will be observed, safety assessment efforts have been concentrated on the computer hardware due to the computer configuration.

The safety assessment (verification/validation) process as used by the RTRI consists of activities conducted in two main phases of development: system design and testing.

3.17.1.1 System Design Stage - The first activity in the system design stage is the conduct of a preliminary safety analysis to identify those portions of the system that are critical to safety. Then, a safety analysis is conducted on those portions. This begins with the conduct of a failure modes and effects analysis on those hardware circuits that are required to be fail-safe. It also involves the conduct of a fault tree analysis (FTA) on those subsystems where hardware errors or operator actions could lead to a fatal failure. A third analysis technique involves the use of a fault injection simulator. In this method, a simulation program is executed on a personal computer or experimental test set. Faults such as stuck outputs or input-output shorts are injected at the gate level, and the resulting effects are observed.

Another activity in the design stage is a quantitative evaluation of system safety (and availability) based upon a Markov Process Model technique. The unsafe failure rate of computer-based signalling systems in Japan is typically required to be on the order of 1×10^{-9} failures per hour. On the other hand, mean system down time in a safe state is required to be less than 30 minutes per every 10 years.

Still another design stage activity involves a design review. This is done to examine the hardware and software specifications for compliance with requirements and other guidelines.

3.17.1.2 Test Stage - A fault injection test is done on the prototype of a newly developed fail-safe circuit. In addition, another hardware test is done using a computer and test program in combination with the fail-safe circuit. Hardware is then tested against environmental stresses (e.g., temperature, humidity, vibration), even beyond the specifications.

Following data tests and tests of software modules, system functional tests are performed. These utilize a simulator to simulate the signalling devices and the train itself. Error handling tests are performed in which harJware faults or input data errors are injected. The system is also tested for noise using a noise simulator and surge generator. Finally, a monitor run test is performed in which a new system is installed in the field and the control outputs are compared with those of existing equipment. This test usually runs from several months to one year.

3.17.2 SMILE Interlocking System Design/Assessment

A safety critical computer-based interlocking system referred to as SMILE (safety multiprocessor system for interlocking equipment) was developed/assessed in part by the RTRI and is being used by the Japan Railways. In this system, safety critical interlocking functions are performed by a "fail-safe microcomputer" system in a triple-modular-redundant (TMR) configuration. Miscellaneous functions are performed by other microcomputers in the system. Each of the three TMR computers are connected to a system bus referred to as SMILE-BUS. To detect errors/failures in the system, a fail-safe output voting circuit and fault detector are employed.

According to the RTRI literature, the safety of the system is essentially provided by the hardware structure. However, correctness of the software is achieved through structured design and software testing. This is facilitated by dividing the software into many simple functional modules which are arranged in a hierarchical structure. Software integrity is tested in a dual path arrangement in which all combinations of interlocking conditions are compared against a check list generator. To enhance safety, certain other methods are used in the software design. This includes such techniques as safe-side assignment of information, asymmetrical design, time redundancy and integrity checks of input data.

The safety assessment (V&V) process applied to SMILE included the analysis of failure modes for each circuit and the conduct of safety tests on some prototype circuits to confirm analysis results. In addition, a Markov process model was used to perform a quantitative evaluation of unsafe (and safe) error rates.

3.18 EAST JAPAN RAILWAYS

The East Japan Railways (EJR), in Tokyo, Japan, is the largest of seven railway authorities (six passenger and one freight) within the Japan Railway (JR) Group. The JR Group itself was established in 1987, just after privatization of the Japan National Railways.

A brief response was received from the Safety Research Laboratory of EJR which indicated that the EJR does conduct some verifications and validations to ensure the safety of computer-based signalling equipment. However, neither a summary of the process nor further details could be furnished at the time of this report.

Personnel within the Safety Research Laboratory indicated that further details could be obtained from the International Division of the EJR. However, the International Division was not able to supply any further information at this time.

3.19 NIPPON SIGNAL

The Nippon Signal Company, located in Tokyo, Japan, has been developing and supplying signalling systems in Japan and elsewhere for over 60 years. Various departments with Nippon Signal perform safety verification and validation activities at different stages in a system's development cycle.

Details of the V&V activities were not available for this report. However, personnel in the Signal Engineering Department of Nippon Signal indicated that details could be provided at a later date.

A very general overview of the V&V activities performed on safety critical computer-based systems is provided below.

3.19.1 Verification

The following activities are performed on hardware:

- Plan Review -- conducted by Design Department Manager when requested specifications are satisfied (prepared)

- Conceptual Review -- conducted by Design Department Manager when conceptual design is completed

- Basic Design Review -- conducted jointly by Design and Inspection Departments when basic design is completed

- Production Design Review -- conducted jointly by Design and Inspection Departments when production design is completed

- Product Review – conducted jointly by Design and Inspection Departments when all adjustments (modifications) and inspections are completed.

Software verification activities include:

- Design Reviews – conducted by Design Department Manager when systems design and basic program design are completed
- Walk-through – conducted by design personnel when detailed design is completed.

3.19.2 Validation

The following validation activities are performed on hardware and software as indicated:

- Product Workmanship Review – conducted by Inspection Department when production hardware is completed
- Hardware Testing – conducted by Inspection Department when hardware/software integration is completed
- Software Testing – conducted by Inspection Department for the purpose of testing specific functions allocated to the software.

Products are classified by safety level, and validation activities on high-level (safety critical) products include the conduct of special safety-related "breakdown tests" on prototypes.

Nippon Signal is in the process of reviewing and revising their V&V process and activities.

3.20 INTERNATIONAL ELECTROTECHNICAL COMMISSION

The International Electrotechnical Commission (IEC), with headquarters in Geneva, Switzerland, issues a wide variety of standards for many different applications. Several of these standards are identified and addressed in other sections of this report (e.g., Nuclear Regulatory Commission, Medical Industry). However, there are two draft industrial standards which not only pertain to the safety of computer-based systems/equipment, but also are serving as a basis for standards in the railway industry, particularly in Europe. For example, the new railway safety standards being prepared by working groups within the CENELEC organization for the European Community (EC) are basing much of their standards on the content of the IEC draft standards. The two main IEC documents (latest available versions) being addressed here are as follows:

- IEC 65A (Secretariat) 122, "Software for Computers in the Application of Industrial Safety Related Systems," version dated November 1991, and

- IEC 65A (Secretariat) 123, "Generic Aspects: Functional Safety of Electrical/Electronic/Programmable Electronic Safety Related Systems, Part 1: General Requirements," version 7, dated 1992.

The first addresses software, while the latter deals with more general system/hardware aspects. Indications are that these two standards will soon be merged into a single standard composed of three parts: generic, electrical/electromechanical and software. Brief overviews are provided below for the two draft standards.

3.20.1 IEC 65A (Secretariat) 122

This draft standard (88 pages) presents requirements for achieving safety integrity of software in computer-based systems. It applies to the development and assessment of that software, and is to be used in conjunction with the other draft standard (i.e., 65A 123). The general steps to achieve software integrity, as outlined in this draft, are as follows:

- Obtain or produce the computer system safety requirements specification
- Identify the safety functions allocated to the software
- Allocate software integrity levels to software safety functions
- Determine more detailed software requirements and the software architecture to be used
- Design, develop, and verify/test the software according to the Software Quality Plan, software integrity level and software life cycle
- Integrate the software and target hardware
- Validate the software
- Reconsider appropriate aspects of this standard when software maintenance is performed.

Key sections of this draft of particular interest in this study are as follows: Section 12 (Verification), Section 14 (Software Validation), Section 15 (Assessment), and Section 16 (Quality Assurance). Detailed descriptions of various design/assessment techniques as well as their advantages/disadvantages are provided in Annex B of the draft.

3.20.1.1 Verification - Verification of the software is to be carried out by an independent organization in order to test and evaluate the products of different phases of development. The objective is to ensure correctness and consistency with products that serve as inputs to the different phases. The extent of the verification is determined by the assigned integrity level. Suggested verification techniques (e.g., formal proof, probabilistic testing, static analysis, dynamic testing, metrics) are identified.

3.20.1.2 Software Validation - The purpose of validation here is (for an independent assessor) to test the integrated system to ensure compliance with software requirements. Suggested validation techniques include probabilistic testing, simulation/modeling and functional/black-box testing.

3.20.1.3 Software Assessment - The purpose of this activity is to evaluate the life cycle processes and products to determine that the software has the proper integrity level and is fit for its intended application. This amounts to a review of all safety-related activities and results that comprise the development life cycle.

3.20.1.4 Quality Assurance - The purpose of this activity is to identify, monitor and control all technical and managerial activities that are necessary to ensure software safety. This includes ensuring compliance with a quality assurance system such as described in the ISO 9000 series of standards.

3.20.2 IEC 65A (Secretariat) 123

This draft (140 pages) describes a general approach for performing system safety activities relative to an overall safety life cycle and another life cycle referred to as the E/E/PES (electrical/electronic/programmable electronic system) life cycle, which is more directed to the specific safety critical portions of the system.

3.20.2.1 Overall Safety Life Cycle - Major phases of the overall safety life cycle with summaries of their objectives are as follows:

- Concept -- develop understanding of the equipment under control

- Overall System Definition -- determine scope of the subsequent hazard and risk analysis including equipment to be included

- Hazard and Risk Analysis -- determine potential hazards and associated risks

- Overall Safety Requirements -- develop Overall Safety Requirements Specification including Overall Functional Requirements Specification, Overall Safety Integrity Requirements Specification and Overall Safety Requirements Report

- Allocation of Safety Requirements -- allocate target safety requirements to appropriate system portions

- Overall O&M Strategy -- develop plans to ensure safety is addressed in operation and maintenance activities

- Overall Validation Planning -- develop Overall Safety Validation Specification to ensure validation of total system
- Safety-Related System Realization -- create safety-related systems from requirements specifications
- Other Technology Safety-Related System Realization -- create "other technology" safety-related systems from requirements specifications
- External Risk Reduction Facilities Realization -- create appropriate risk reduction facilities
- Overall Installation -- install safety-related systems and risk reduction facilities
- Overall Safety Validation -- validate that the combination of safety-related systems and risk reduction facilities meets the Overall Safety Validation Specification
- Overall O&M -- operate and maintain combination of safety-related systems and risk reduction facilities to ensure safety
- Overall Modification and Retrofit -- ensure safety following modifications and retrofitting activities
- Decommissioning -- ensure safety during decommissioning process.

Requirements are described for each of these phases. It should be noted that a hazard analysis and risk assessment as well as safety validation are key parts of this cycle. The hazard analysis and risk assessment activity, conducted to identify hazards and assign risk, is a continuing process throughout the life cycle. Guidance is provided as to how to determine risk and integrity levels.

Although not strictly identified in the overall safety life cycle, verification and functional safety assessment tasks are also required throughout development. The purpose of the continuing verification task is to ensure via reviews, analyses and/or testing that deliverables comply with objectives and requirements for a given activity. The functional safety assessment is to ensure that all safety-related systems and risk reduction facilities are functionally safe. This is also a continuing process throughout the life cycle.

3.20.2.2 E/E/PES Safety Life Cycle - This life cycle defines the activities for ensuring the functional safety of the safety-related portions of the system. This cycle includes the following activities:

- Safety Requirements Specification
- Functional Requirements Specification

- Safety Integrity Requirements Specification
- Design
- Validation Planning
- O&M Procedures
- Implementation
- Safety Validation.

The safety validation activity is to ensure that the safety-related system portions meet the Safety Requirements Specifications. Guidance in the selection of techniques to use for this activity, and particularly, how to select techniques based upon the assigned integrity levels, is provided in Annexes to the document. Examples of techniques which may be needed are: inspection, walkthrough, static or dynamic analysis, simulation, FMEA, worst case analysis, surge immunity testing and others. Verification and functional assessment activities are also to be a part of this E/E/PES life cycle as they were for the overall system safety life cycle.

3.21 INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS

The Institute of Electrical and Electronics Engineers (IEEE), with headquarters in New York, New York, has been issuing numerous standards, guidelines and recommended practices for many years pertaining to a wide variety of systems, applications and topic areas. Much attention has been given to topics pertaining to software including terminology, design, implementation, verification and validation (V&V), quality assurance, documentation and software engineering. Many of the resulting standards have been adopted by the American National Standards Institution (ANSI).

Significant attention in recent years has been directed to the development and maintenance of software. Some examples of existing ANSI/IEEE standards pertaining to this topic are as follows:

- ANSI/IEEE 730-1984, Software Quality Assurance Plans
- ANSI/IEEE 828-1990, Standard for Software Configuration Management Plans
- ANSI/IEEE 829-1983, Standard for Software Test Documentation
- ANSI/IEEE 830-1984, Guide for Software Requirements Specification
- ANSI/IEEE 1012-1986, Standard for Software Verification and Validation Plans

- ANSI/IEEE 1016-1987, Recommended Practice for Software Design Descriptions

- ANSI/IEEE 1028-1988, Standard for Software Reviews and Audits.

In fact, many of these and other standards are published as a set in a periodic publication entitled "IEEE Software Engineering Standards Collection."

One of the above documents, namely ANSI/IEEE 1012-1986, deals with the subject of software V&V, and is the primary existing IEEE V&V-related document. In general, the purpose of the document is to define uniform and minimum requirements for the format and content of Software Verification and Validation Plans (SVVPs). The terms verification and validation are used as defined in the glossary of this overall project report. To better understand their usage in this discussion, their definitions are repeated below:

- Verification — The process of determining whether or not the products of a given phase of the software development life cycle fulfill the requirements established during the previous phase

- Validation — The process of evaluating software at the end of the software development process to ensure compliance with software requirements.

It should be emphasized that, although the V&V process as presented in this document applies to both critical (i.e., safety or financial/social loss-related) and non-critical software, its use is not intended to, by itself, ensure the safety of the software.

For this reason, the IEEE P1228 Working Group is in the process of developing another standard pertaining to software safety plans. It has been recognized that further efforts are needed, beyond the existing V&V process and quality assurance measures, to help ensure the safety of software. The standard being prepared, namely P1228, "Standard for Software Safety Plans," was in draft form (Draft J) at the time of writing this report. The intent of P1228 is to define and describe the mandatory elements of a Software Safety Plan, addressing the entire life cycle of software development. The intent will be for this new standard to be used in conjunction with other existing ANSI/IEEE software development/quality assurance standards to produce and demonstrate the safety of the software.

Overviews of the above two relevant software-related documents (i.e., ANSI/IEEE 1012-1986 and P1228) are provided below, with emphasis on the latter. Generic hardware/system related safety verification/validation standards are not addressed in this section. In fact, it appears that ANSI/IEEE has not issued generic standards pertaining specifically to system/hardware safety. As mentioned earlier, IEEE's recent focus in the area of computer system safety has been on software aspects. One example of a more application-oriented ANSI/IEEE standard that does address system and hardware as well as software safety aspects is the document (and new revised draft) ANSI/IEEE-7-4.3.2-1982. This is discussed in more detail elsewhere in this report.

3.21.1 ANSI/IEEE Std 1012-1986

As indicated above, this standard (which applies to critical and non-critical software) defines requirements for the format and content of Software Verification and Validation Plans (SVVPs). It describes the minimum V&V tasks and the inputs and outputs that should be included for critical software. It also provides guidance on tailoring the SVVP to fit a particular application.

V&V is to be performed in parallel with software development, and should help ensure the following:

- Errors are detected and corrected as early as possible
- Project risk, cost and schedule effects are reduced
- Overall software quality/reliability is enhanced
- Management visibility of the software process is improved, and
- Proposed changes (and their effects) can be quickly assessed.

The plan is to comprise the following seven sections:

1) Purpose
2) Referenced documents
3) Definitions
4) V&V Overview
5) Life-Cycle V&V
6) Software V&V Reporting
7) V&V Administrative Procedures.

Section 5 of the plan is to describe in detail the tasks that will be performed throughout the software life cycle. Required information for each task includes the reason for the task, methods and criteria to be used, source and format of inputs and outputs, schedule, required resources, risks/assumptions and roles/responsibilities for those performing the task. The recommended minimum V&V tasks for critical software applications are listed in Table 3-3.

TABLE 3-3. MINIMUM V&V TASKS FOR CRITICAL SOFTWARE APPLICATIONS ACCORDING TO ANSI/IEEE 1012-1986

Activity/Phase	Task
Management of V&V	Software V&V Plan Generation Baseline Change Assessment Management Review Review Support
Concept Phase V&V	Concept Documentation Evaluation
Requirements Phase V&V	Software Requirements Traceability Analysis Software Requirements Evaluation Software Requirements Analysis System Test Plan Generation Acceptance Test Plan Generation
Design Phase V&V	Design Traceability Analysis Design Evaluation Design Interface Analysis Component Test Plan Generation Integration Test Plan Generation Test Design Generation
Implementation Phase V&V	Source Code Traceability Analysis Source Code Evaluation Source Code Interface Analysis Source Code Documentation Evaluation Test Case Generation Test Procedure Generation Component Test Execution
Test Phase V&V	Test Procedure Generation Integration Test Execution System Test Execution Acceptance Test Execution
Installation and Checkout Phase V&V	Installation Configuration Audit V&V Final Report Generation
Operation and Maintenance Phase V&V	Software V&V Plan Revision Anomaly Evaluation Proposed Change Assessment Phase Task Iteration

3.21.1.1 NIST Endorsement - One responsibility of the National Institute of Standards and Technology (NIST), located in Gaithersburg, Maryland, is to develop standards and guidelines in a variety of technological areas for use by the Federal Government. In the area of software V&V, NIST has endorsed the above IEEE standard (i.e., ANSI/IEEE 1012-1986). This endorsement and other information on software V&V is addressed in the following three key NIST documents:

- FIPS PUB 101, "Guideline for Lifecycle Validation, Verification and Testing of Computer Software," June 6, 1983

- FIPS PUB 132, "Guideline for Software Verification and Validation," 1987, and

- NIST Special Publication 500-165, "Software Verification and Validation: Its role in Computer Assurance and Its Relationship with Software Project Management Standards," 1989.

These documents suggest that software V&V ensures software quality and optimum performance, and also helps to produce safe, secure and reliable software programs.

3.21.2 P1228 (Draft)

The intent of this draft standard (P1228, "Standard for Software Safety Plans," Draft J) is to establish the minimum elements (e.g., processes and activities) of a Software Safety Plan (SSP) that could subsequently be used to improve the safety of software in safety critical applications. It is acknowledged that the plan must address safety in the context of the entire system, and must address software's interfaces with its associated hardware, environment and operators. While the intent is to establish a set of minimum requirements for the SSP, additional software-related safety requirements are encouraged.

The standard is organized into the following six sections:

- Purpose – defines the purpose and scope of the plan, safety goals and objectives, and acceptable risks

- Definitions, Acronyms and References

- Software Safety Management – describes the organization, schedule, resources, responsibilities, tools, techniques and methodologies used in the development of the software

- Software Safety Analyses – defines the various safety analyses and tests to be performed during the software development process

- Post Development – defines requirements (e.g., training, maintenance) to ensure continued safety after deployment

- Software Safety Design Analysis – verifies that the safety-critical portions of the design meets safety critical requirements; some possible analysis methods include:

    - Logic analysis – evaluates equations, algorithms and control logic
    - Data analysis – evaluates usage of data items
    - Interface analysis – evaluates interfaces with other system components
    - Constraint analysis – evaluates safety-related "real world" restrictions (e.g., unacknowledged interrupt)
    - Functional analysis – ensures each safety critical requirement is covered
    - Module analysis – examines non-safety critical modules for possible hazards
    - Evaluation of environment based upon timing and sizing estimates
    - Reliability predictions for software modules

- Software Safety Code Analysis – ensures correct implementation (of safety critical portions of software) in the code; possible analysis methods include:

    - Logic analysis – evaluates sequence of operations in the code
    - Data analysis – evaluates data usage
    - Interface analysis – evaluates module compatibility
    - Constraint analysis – ensures program operates within constraints of requirements, design, and the target computer
    - Programming style analysis – ensures code follows approved programming guidelines
    - Non-critical code analysis – examines non-critical code portions for possible hazards
    - Timing and sizing analysis – checks for timing and sizing problems in code

- Software Safety Test Analysis – demonstrates via testing that safety requirements have been correctly implemented and software functions safely in its environment; some possible testing includes:

    - Computer software unit level testing – demonstrates correct execution of critical software
    - Interface testing – demonstrates software units operate together correctly
    - Computer software integration item testing – demonstrates correct performance of one or more software components
    - System level testing – demonstrates performance within overall system

- Stress testing — evaluates software execution under abnormal circumstances (e.g., abnormal inputs)
- Regression testing — examines software changes for introduced hazards

- Software Safety Change Analysis — demonstrates that changes do not adversely impact safety; involves all types of changes (e.g., assumptions, specifications, design, code, test plans and many other areas); in general, the plan is to describe how the impact of the change will be determined, and what tests/analyses will be used to ensure safety is not adversely affected; details are not specified — they are left to the developer.

3.22 DEPARTMENT OF DEFENSE

The primary standard for the Department of Defense (DOD) in the area of system safety is MIL-STD-882C, "System Safety Program Requirements," dated January 19, 1993. It provides requirements for developing and implementing a system safety program that applies to the entire life cycle of a system. These requirements include various safety assessment activities directed to identifying hazards and eliminating or reducing risks associated with all DOD systems (and facilities) including those incorporating computers in safety-critical applications.

This standard, to be discussed in more detail later in this section, supersedes the long-standing MIL-STD-882B and a 1987 revision (Notice 1) which was issued to address software safety issues. This new system safety standard (i.e., 882C) also supersedes another DOD system safety standard entitled "System Safety Standard for Space and Missile Systems," MIL-STD-1574 (15 August 1979). MIL-STD-1574 addresses system safety, but with very little emphasis on software activities conducted in conjunction with system development. MIL-STD-882C, on the other hand, very much addresses software (and hardware) safety activities that are to be integrated into the development process.

There are two other key DOD standards that, while not specifically directed (by themselves) to ensuring system or software safety, do pertain to improving the overall quality of the software and its associated documentation. They are:

- DOD-STD-2167A, "Defense System Software Development," and

- DOD-STD-2168, "Defense System Software Quality Program"

The former presents software development requirements including V&V activities, and the latter presents requirements for an overall software quality program. Both of these standards are typically used in conjunction with MIL-STD-882C. Brief overviews of these are also provided below, but in less detail than the primary safety standard (i.e., 882C).

In addition to the above standards, there are two other safety-related DOD standards which apply to nuclear weapon systems. Those are:

- AF Regulation 122-9, "The Nuclear Surety Design Certification Program for Nuclear Weapon System Software and Firmware," and

- AF Regulation 122-10, "Nuclear Surety Safety Design Criteria for Nuclear Weapon Systems."

The former describes the safety certification process for computer-based nuclear weapon systems. It specifies the need for an independent V&V effort as well as a Nuclear Safety Cross Check Analysis. The latter document is typically used in conjunction with 122-9, and deals primarily with the design of weapon system handling mechanisms.

The Trusted Computer Security Evaluation Criteria (TCSEC) or "Orange Book," DOD 5200.28.STD, deals primarily with security as opposed to safety verification and validation, and thus is not addressed in this study.

One other area which is currently receiving great attention in the DOD and other industries is that of standardization in software development process assessment. Although not directly related to safety verification or validation, it does relate to the overall software development process. For interest, three such standardization initiatives are identified here. One is the Software Engineering Institute's Capability Maturity Model (CMM). The CMM, described in the document "Capability Maturity Model for Software" (CMU/SEI-91-TR-24), presents a method for assessing the capability of a contractor's software development process as well as guidance for a contractor in improving their overall software engineering process. A second initiative is the TRILLIUM model for Telecom Software Product Development Capability Assessment, inspired by the CMM, and developed in part by Bell Canada. A third initiative involves a joint effort by the International Standards Organization (ISO) and the International Electrotechnical Commission (IEC). This effort is directed to developing an international standard for software process assessment. It is being called SPICE (Software Process Improvement and Capability dEtermination), and will be used for software development capability determination as well as software development process improvement.

3.22.1 MIL-STD-882C

This military standard describes the requirements for developing and implementing a system safety program which would allow for the identification of hazards and the elimination and/or control of mishaps (e.g., death, injury, damage to equipment/environment). The standard presents the requirements in such a manner that they can be tailored and selected by the procuring agency, depending upon the application. Significant guidance on tailoring as well as implementing the requirements is provided in Appendix A of the standard. That appendix also provides a discussion on software hazard risk assessment. Other appendices deal with such topics as relating requirements to specific life cycle phases and documentation.

Following definitions and general requirements (such as general information on conducting hazard analyses), there is a section in 882C on Detailed Requirements. This section presents detailed requirements for the system safety program in the form of specific tasks. These tasks are separated into four main task sections: Section 100-Program Management and Control, Section 200-Design and Integration, Section 300-Design Evaluation, and Section 400-Compliance and Verification. Brief overviews of these are provided below. It should be pointed out that this standard presents information in the form of more general requirements and, in most cases, does not require specific analysis/testing techniques. In most instances, these are left to the discretion of the managing authority with recommendations from the contractor.

3.22.1.1 Section 100-Program Management and Control - This section describes the various requirements for establishing and managing the system safety effort. A total of seven tasks (Task 101 - 107) address such topics as establishing a system safety program plan (Task 102), conducting system safety program reviews/audits (Task 104), and establishing a hazard tracking system (Task 106).

3.22.1.2 Section 200-Design and Integration - This section describes the various tasks and analyses that could be conducted during the design process to identify and eliminate/control potential hazards in the system. The nature/purpose of each of the seven tasks is summarized below:

- Task 201-Preliminary Hazard List (PHL) – examine the system shortly after concept definition and prepare a preliminary hazard list which identifies potential hazards

- Task 202- Preliminary Hazard Analysis (PHA) – conduct a PHA early in the design effort to identify safety critical areas, assess hazards and identify possible hazard control actions

- Task 203-Safety Requirements/Criteria Analysis (SCRA) – relate the identified hazards to the system design and develop design requirements to reduce the risk of those hazards

- Task 204-Subsystem Hazard Analysis (SSHA) – identify all components and equipment that could result in a hazard or whose design does not satisfy safety requirements; includes determination of such things as single point and common mode failure effects, and the correct translation of design requirements from top level specifications to detailed specifications; if DOD-STD-2167 and DOD-STD-2168 or other standards are being used for software development, the output of each development phase is to be used in evaluating the software contribution to the SSHA

- Task 205-System Hazard Analysis (SHA) – identify hazards and assess risk of total system design including software and subsystem interfaces; as above, outputs of the software development process pnases are to be used in evaluating software's contribution to the SHA

- Task 206-Operating and Support Analysis (O&SHA) – identify and evaluate hazards introduced by operational and support procedures; includes hazards introduced by human errors

- Task 207-Health Hazard Assessment (HHA) – identify potential health hazards, evaluate proposed hazardous materials, and propose protective measures.

Specific techniques to be used in many of the above analyses can be directed by the managing authority and/or recommended by the contractor, but in any case, final approval of the techniques used must be given by the managing authority.

3.22.1.3 Section 300-Design Evaluation - The following three tasks are described:

- Task 301-Safety Assessment – conduct a safety assessment of the system being acquired in order to 1) identify safety features of the hardware, software and overall system design, and 2) identify procedural, hardware and software related hazards that could be present in the system; the methodology used to classify and rank hazards as well as all analysis/test results relating to the identification of hazards should be documented

- Task 302-Test and Evaluation Safety – make sure the contractor test and evaluation safety activities recommend actions and assess actions taken to reduce, correct, or control (at a minimum) catastrophic and critical level hazards

- Task 303-Safety Review of Engineering Change Proposals (ECPs), Specification Change Notices (SCNs), Software Problem Reports (SPRs), and Requests for Deviation/Waiver – conduct analyses on all changes, problem reports, etc., to determine impact upon safety.

3.22.1.4 Section 400-Compliance and Verification - The following tasks are described:

- Task 401-Safety Verification – perform tests, demonstrations, modeling or other techniques to verify compliance of safety critical hardware, software and procedures (e.g., emergency procedures) with safety requirements.

- Task 402-Safety Compliance Assessment – conduct an assessment to verify compliance of design and other procedures with military, federal, national, international and industry codes, standards and specifications and evaluate safety risk; may include hazard analysis, drawing/procedural reviews and equipment inspections.

Tasks 403 and 404 address some additional safety aspects of systems dealing with explosive devices, and thus are not discussed here.

3.22.2 DOD-STD-2167A

This document, entitled "Defense System Software Development," contains requirements for the acquisition, development and support of software (both safety and non-safety related). It is to be used in conjunction with other documents (e.g., MIL-STD-1521, "Technical Reviews and Audits for Systems, Equipment and Computer Software") to improve the overall quality of the software and its associated documentation. As mentioned earlier, the outputs of the various development phases are to be used as inputs to the risk assessments described in MIL-STD-882C.

DOD-STD-2167A describes a menu of deliverables, analyses/tests, reviews and audits that could be performed and a set of management practices that could be utilized. It does not impose a particular software development or design methodology. Rather, it encourages the contractor to select development methods to best meet overall contractual requirements. Further, the requirements discussed in the standard are to tailored by the managing authority and contractor for a specific application. Guidance for tailoring can be found in MIL-HDBK-287, "A Tailoring Guide for DOD-STD-2167A, Defense System Software Development."

General and detailed requirements are provided for six major functional categories: software development management, software engineering, formal qualification testing, software product evaluations, configuration management and transition to software support. Detailed requirements are further separated into the following areas:

- Software Requirements Analysis
- Preliminary Design
- Detailed Design
- Code and Computer Software Unit Testing
- Computer Software Component Integration and Testing
- Computer Software Configuration Item Testing, and
- System Integration and Testing.

The document does not include a requirement for safety analysis, but merely indicates that the analysis is to ensure that the software requirements, design and operating procedures minimize the potential for hazardous conditions during the mission. Specific safety analysis techniques are not identified. The requirements of MIL-STD-882C are to ensure the safety of the overall system including the software.

The standard also addresses interface requirements should an independent V&V contractor be involved in the software development process. Guidance for the IV&V effort itself can be obtained from the document AFSC/AFLC Pamphlet 800-5, "Software Independent Verification and Validation."

There is, in fact, an effort underway to revise DOD-STD-2167A and DOD-STD-7935A ("DOD Automated Information Systems Documentation Standards") and merge them into one document designated MIL-STD-SDD. The currently available version of this document (dated December 22, 1992) is undergoing review by a special committee.

3.22.3 <u>DOD-STD-2168</u>

This standard, "Defense System Software Quality Program," presents both general and detailed requirements for a software quality program that could be applied during the acquisition, development, and support of software. The standard is to be used in conjunction with other standards such as DOD-STD-2167A. According to this standard (i.e., DOD-STD-2168), software quality is the ability of the software to satisfy its specified requirements.

3.23 FEDERAL AVIATION ADMINISTRATION

The Federal Aviation Administration (FAA), with headquarters in Washington, D.C., has established regulations pertaining to Airworthiness Standards for Transport Category Airplanes. These are found in 14 CFR Part 25. Paragraph 25.1309 (Equipment, Systems and Installations) specifies qualitative requirements which are applicable to the topic of safety verification/validation. The major requirements of interest here are found in sections (a) through (e) of 25.1309. They can be summarized as follows:

(a) Equipment, systems and installations must be designed to perform their intended functions under any foreseeable operating conditions.

(b) Systems and components must be designed so that 1) the occurrence of any failure which would prevent the safe flight and landing of the airplane is extremely improbable, and 2) the occurrence of any other failure which would reduce the capability of the airplane or crew to cope with the adverse condition is improbable.

(c) Warnings must alert the crew to unsafe system operating conditions, and allow them to take corrective action; also, associated systems and controls should be designed to minimize crew errors.

(d) Compliance with section (b) must be shown via analysis or testing as appropriate. Analyses must consider the following: 1) possible modes of failure, 2) probability of multiple and undetected failures, 3) resulting effects on airplane and occupants, and 4) crew warning cues, corrective actions and the capability of detecting faults.

(e) Power sources must supply power to "essential loads" in a reliable manner (as defined in 25.1309).

Further, 14 CFR, Part 21 (Certification Procedure for Products and Parts) establishes requirements for certifying aircraft and associated components. In general, manufacturers must submit evidence (e.g., specifications, analyses, test results) of compliance with all applicable airworthiness requirements and any other requirements imposed by the FAA. The certification authority (typically, Aircraft Certification Offices or ACOs) determines compliance with requirements and reports to the appropriate FAA Directorate. Software based systems/equipment are certified in essentially the same manner. However, in these instances, the certification authority assesses additional documentation prepared by the manufacturer which includes a Plan for Software Aspects of Certification and Software Accomplishment Summary.

Guidance for interpreting and complying with (safety-related) airworthiness standards for systems and equipment certification (Paragraph 25.1309), including computer-based systems, is provided in two main documents as follows:

- FAA Advisory Circular, AC 25.1309-1A. "System Design and Analysis," dated June 21, 1988, and

- RTCA/DO-178B, "Software Considerations in Airborne Systems and Equipment Certification," dated December 1, 1992.

AC 25.1309-1A deals with system and hardware aspects of certification while DO-178B addresses software. An overview of the nature and content of these two documents (with emphasis upon safety verification/validation) is provided below. It should be emphasized that these represent guidelines and not mandatory standards for the FAA.

In addition to the above existing documents, further work is being conducted in this area by various committees within the Society of Automotive Engineers (SAE). Their recent work has resulted in at least two pertinent draft Aerospace Recommended Practice (ARP) documents dealing with safety verification/validation/assessments for FAA systems and equipment. These are:

- ARP 4754, Draft 23C, "Systems Integration Requirements," dated January 19, 1993, and

- ARP 4761, Draft 4, "Safety Assessment Guidelines for Civil Airborne Systems and Equipment," dated February 26, 1993.

Both provide guidance for systems aspects in the context of certification, and are intended to be used (when released in final form) in conjunction with other documents such as AC 25.1309-1A and DO-178B. Brief overviews of these documents are also provided on the following pages.

3.23.1 AC 25.1309-1A

This document describes various possible means for complying with Paragraph 25.1309 of the airworthiness standards, and in particular with the analysis/assessment requirements of sections (b), (c), and (d). It applies to system and hardware aspects, but does not address software. The reader is referred to RTCA/DO-178A or later revisions (i.e., DO-178B) for software assessment guidance.

According to Paragraph 25.1309, Sections (b) and (d), analyses and testing (when necessary) are required to show compliance with the requirements. The objective is to ensure an orderly and structured evaluation of the effects of failures and other events on safety.

It is first suggested that a qualitative functional hazard analysis (FHA) be conducted to identify and classify potentially hazardous failure conditions. Such conditions are to be classified by severity (i.e., minor, major or catastrophic), and will have one of three probability classifications (i.e., probable, improbable or extremely improbable). The analysis is to show that minor conditions can be classified as probable. Then, further analysis is necessary to show that major conditions are improbable and catastrophic conditions are extremely improbable.

In analyzing failure conditions classified as major, two considerations are described. First, for relatively simple (non-complex and non-redundant) systems, it may be possible to evaluate them in light of satisfactory performance on other existing systems. This may also be supported by a failure mode and effects analysis (FMEA), fault tree, or reliability block diagram analysis for more complex systems. Second, for redundant systems, isolation between channels and satisfactory reliability should be shown. An FMEA, fault tree, or reliability block diagram analysis may again be used.

When dealing with catastrophic conditions, a "very thorough" safety assessment is suggested. Sometimes it may be sufficient to assess the system conditions in light of a similar system in operation -- showing that no catastrophic conditions have occurred. However, it is most likely that a combination of qualitative and quantitative analyses be performed. Suggested qualitative analyses include design appraisal, installation appraisal, FMEA, fault tree, and/or reliability block diagram analysis. A (quantitative) probability analysis could be used to supplement the qualitative analyses. This could involve the conduct of an FMEA, fault tree, or reliability block diagram with numerical probabilities. Extremely improbable failure conditions are defined as those having a probability of 1×10^{-9} or less. Again, tests may be conducted as appropriate to support the analyses.

Some guidance is also provided on addressing latent failures, environmental conditions, and operational/maintenance aspects.

3.23.2 RTCA/DO-178B

The RTCA, originally the Radio Technical Commission for Aeronautics, and now the RTCA, Inc. (Requirements and Technical Concepts for Aviation), was established in 1980 to develop software practices that would support the development of software-based airborne systems and equipment. The document DO-178A was released in 1985 to address software-based systems. In 1989, a special committee (SC-167) was formed at the request of the FAA to revise DO-178A and take into account changes in software technology. The committee worked closely with a similar organization in Europe known as the European Organization for Civil Aviation Equipment (EUROCAE). Specific areas of interest were:

- Documentation Integration and Production
- System Issues
- Software Development
- Software Verification
- Software Configuration Management and Software Quality Assurance.

In 1992, DO-178B was released by the FAA, and concurrently, a document referred to as ED-12 was released by EUROCAE.

This document (i.e., DO-178B) provides guidance for determining compliance of airborne system software aspects with airworthiness requirements. It describes objectives for the overall software life cycle, means of achieving those objectives, and evidence that indicates the objectives have been satisfied. This includes software verification and validation activities as they pertain to the overall software development process. The relationship of the software life cycle to the system life cycle is also described.

The first part of the document discusses various system and general software aspects and their relationships to the software development process. This includes such topics as information flow between system and software processes, system architectural considerations, partitioning, use of multiple version dissimilar software, safety monitoring and use of user-modifiable and field-loadable software. Perhaps one of the most important portions of this material for this study is the section on software level determination. Five failure categories (for a system) are defined (i.e., catastrophic, hazardous/severe-major, major, minor and no effect), based upon the severity of a failure on the aircraft and its occupant. Then, five software levels are defined (i.e., level A through E) in light of these failure conditions. Level A, the most critical, could result in a catastrophic failure in the aircraft.

3.22.2.1 Software Life Cycle - The software life cycle is described as being composed of three main processes: software planning, software development, and integral processes (which includes software verification activities). Software planning defines the means for satisfying system and airworthiness requirements. Software development produces the itself. The integral processes are performed throughout the software life cycle. Since software verification is part of the integral processes (Chapter 6), most attention here is being directed to this chapter.

3.23.2.2 Software Verification Process - Verification, as used in DO-178B, refers to the technical assessment of the results of both the software development and software verification processes. Errors are to be detected and reported during the verification process and removed in the development process. The main objectives of software verification are to verify that:

- System requirements allocated to software have been developed into appropriate high-level requirements

- High-level requirements have been developed into software architecture and appropriate low-level requirements

- Software architecture and low-level requirements have been developed into appropriate source code

- Executable object code satisfies software requirements

- Means to satisfy these objectives are technically correct and complete for each software level.

Software verification is achieved through the conduct of reviews, analyses, development of test cases, and execution of those tests. Reviews and analyses provide assessments of the accuracy, completeness, and verifiability of the software requirements, architecture and source code. Development and execution of the test cases may provide a further assessment of the requirements as well as a demonstration of compliance with them.

3.23.2.2.1 Reviews and Analyses - Reviews and analyses are applied to both the software development and software verification processes. A review may involve an inspection using a checklist or similar aid whereas an analysis is more detailed and may involve looking at the functionality, performance, traceability, or other aspects of a software component. Reviews and analyses may be used to detect and report errors in the software, and/or incompleteness or inaccuracies in development/verification processes or activities. Areas to which reviews and analyses are to be directed and their primary objectives are summarized below:

- High-Level requirements -- ensure they comply with system requirements, are accurate and consistent, are compatible with the target computer, are verifiable, conform to Software Requirements Standards, and are traceable to system requirements; and that algorithms are accurate

- Low-Level requirements -- ensure they comply with high-level requirements, are accurate and consistent, are compatible with the target computer, are verifiable, conform to Software Design Standards, and are traceable to high-level requirements; and that algorithms are correct

- Software architecture – ensure it is compatible with high-level requirements, is consistent and verifieable, compatible with the target computer, and conforms to Software Design Standards; and that software partitioning integrity is confirmed

- Source Code – complies with low-level requirements and software architecture, is verifiable, conforms to Software Code Standards, is traceable to low-level requirements, and is accurate and consistent

- Integration process – output of process is complete and correct; can examine the linking and loading map, and should address incorrect hardware addresses, memory overlaps and missing software components

- Test cases – determine whether testing verifies the implementation of the software requirements, and which code structure was not exercised by the test procedures; additional test cases and/or verification may be needed

- Test procedures and results – ensure test cases were accurately developed into test procedures and expected results; and that test results are correct and discrepancies (between actual and expected results) are explained.

3.23.2.2.2 Testing - Testing has two main objectives: 1) to demonstrate that software complies with requirements, and 2) to demonstrate with a high level of confidence that errors which could lead to unacceptable failure conditions have been removed. It is recommended that software be tested in the target computer and in a "high-fidelity" simulation of its environment, but a target computer emulator may be acceptable in some cases.

Requirements-based test case selection and testing is emphasized. Two categories of test cases are suggested as follows:

- Normal range test cases – directed to normal inputs and conditions, and

- Robustness test cases – directed to abnormal inputs and conditions.

Three types of requirements-based test methods are recommended:

- Hardware/software integration testing – ensures software in target computer will satisfy high-level requirements

- Software integration testing – ensures software components interact correctly with each other and satisfy the software requirements and architecture

- Low-level testing ensures software components satisfy their low-level requirements.

3.23.2.2.3 Software Modifications - Guidance is provided for addressing changes to previously developed software. In general, the area affected by the change should be determined (via perhaps data flow analysis, control flow analysis, timing analysis and/or traceability analysis), and affected areas should be reverified by using reviews, analyses, and testing as described above. Consideration should also be given to a possible change in software level.

3.23.2.3 Other Topics - Considerable discussion is provided on a number of other related topics such as software quality assurance, software configuration management, software life cycle documentation, tool qualification, and use of alternative methods in software verification (e.g., formal methods, exhaustive input testing, multiple version dissimilar software and usage of product service history). Below are brief highlights of how two of these topics are treated in this document.

3.23.2.3.1 Formal Methods - These methods involve the use of formal logic, mathematics and computer-readable languages to develop, improve or verify software. Formal methods are viewed as being equivalent to an exhaustive analysis of a system with respect to its requirements, and complementary to testing.

3.23.2.3.2 Software Reliability - This topic deals with the quantitative probability of the existence of errors in a software component. The view in this document is that currently available methods and associated results do not provide sufficient levels of confidence to warrant usage in software certification applications.

3.23.3 ARP 4754

The purpose of this draft of the recommended practice document ARP 4754, "Systems Integration Requirements," is to provide guidance on demonstrating compliance with airworthiness requirements in airborne, complex, computer-based commercial aircraft systems. Its intent is to address the total system life cycle and show the relationship between system development, functional safety assessments and software life cycle processes. The reader is referred to DO-178B for software aspects and ARP 4761 (summarized in the next section) for guidance on specific safety assessment processes.

The document discusses four major supporting tasks that can be conducted during the system development process; those are:

- Validation of requirements
- Verification of design

- Configuration assurance
- Process management.

The first, validation of requirements, involves ensuring that system requirements and assumptions are sufficiently correct and complete. Various guidelines for determining correctness and completeness are presented, as are various validation techniques. The latter includes functional hazard assessment (FHA), preliminary system safety assessment (PSSA), fault tree, dependence diagram analysis (DDA), failures mode and effects analysis (FMEA), reliability studies, simulation and modeling, testing, reviews, inspections, demonstrations, traceability and similarity methods.

Design verification involves determining that the system as implemented meets the specific requirements. This can be achieved through a combination of reviews, analyses and tests. Reviews involve the use of a checklist or similar aid. Analyses can be used to examine such aspects as the functionality and performance of an item. Three different analysis methods are presented: modeling, coverage summary, and system safety assessment. The latter is directed to the safety aspects of the system, and is addressed in more detail in the overview of document ARP 4761. Testing, as discussed here, is requirements-based, and can be performed on all or parts of the system.

Configuration management involves the practices and procedures used to provide surveillance and control over the design, operation, adjustment, repair or modification of the system.

Process management involves the practices and procedures to select and control the design, manufacturing and regulatory compliance processes.

3.23.4 ARP 4761

ARP 4761, "Safety Assessment Guidelines for Civil Airborne Systems and Equipment," presents safety assessment methods/guidelines for certification of aircraft systems and equipment. It is intended to be used in conjunction with other documentation such as ARP 4754 and DO-178B.

According to this draft recommended practice, the overall safety assessment process is an analytical process which is embedded within the system development process (as described in ARP 4754) and involves the conduct of different analyses at various stages in the development.

3.23.4.1 Functional Hazard Assessment (FHA) - This high-level analysis is conducted during the initial stages of system development to identify and categorize potential hazards (failure conditions) in the aircraft's operating environment. This information is used as input to the PSSA, and helps define conditions which demand further analyses.

3.23.4.2 Preliminary System Safety Assessment (PSSA). The PSSA is an iterative analysis process that is initiated early in the design, beginning with the allocation of aircraft functions. Its purpose is to take information from the FHA, determine contributing factors for the potential hazards/failure conditions identified, and determine how the system will meet the qualitative and quantitative requirements for the potential hazards/failure conditions identified. It identifies protective strategies, taking into account fail-safe concepts and architectural attributes. A recommended method for the PSSA is a fault tree analysis (FTA), but a dependence diagram (DD) or Markov analysis are other possibilities.

Common cause analyses are also to be included in the PSSA. Three aspects of a common cause analysis are as follows:

- Zonal Safety Analysis (ZSA) – ensures equipment installation conforms to safety standards for different zones of the aircraft
- Particular Risk Analysis (PRA) – identifies events outside the system (e.g., lightning, leaking fluids) which could cause hazards
- Common Mode Analysis (CMA) – identifies failures that could compromise redundancy/independence.

3.23.4.3 System Safety Assessment (SSA) - The SSA is a systematic and comprehensive evaluation of the implemented system to demonstrate that safety objectives and requirements from the FHA and PSSA are met. It uses results of the FHA, PSSA, and common mode analyses as well as a failure modes and effects summary (FMES). The FMES describes results of failure mode and effects analyses, and groups failure modes according to expected effects. FMEAs may be used to supplement the fault tree and dependence diagrams.

Essentially, the SSA is an integration of all analyses to verify the safety of the overall system.

3.24 NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

The National Aeronautics and Space Administration (NASA) is an extensive organization with numerous safety-related documents describing policy-wide as well as program-specific (e.g., Space Shuttle, Space Station) requirements and guidelines. The intent here is not to identify/address all of these documents, but rather to focus in on those materials and activities directly related to safety verifications/validations/assessments of computer-based systems, and in particular, existing standards or those under development.

One of the key NASA-wide safety documents is entitled "NASA Safety Policy and Requirements Document," NHB 1700.1 (V1-B). The purpose of this document is to describe the policies, objectives, requirements and guidelines that define NASA's Safety Program. A wide range of safety issues are addressed in this document ranging from basic safety management to high-level safety requirements for system safety, nuclear safety, aviation safety, facility safety, fire safety and others. A brief overview of Chapter 3 (System Safety) is provided later in this section.

Another key document is entitled "Safety Policy and Requirements for Payloads Using The National Space Transportation System," NSTS 1700.7B. The primary purpose of this document is to present a set of requirements that help ensure the safe mission operations of payloads with hazardous potential. Due to the increased usage of computers in safety-related payload applications, a study was initiated in 1990 by the NASA/Johnson Space Center Engineering Directorate to revise the requirements in NHB 1700.7B to better address computer controlled systems. This study, documented in the Final Report entitled "The Computer Control of Hazardous Payloads" (July 24, 1991), resulted in a recommendation for an independent organization to perform a hazard analysis of the system and a safety compliance audit of the development organization products. There is an effort underway at Johnson Space Center to generate a more concise requirements document for the development, verification and validation of computer-based systems in payload applications. It will be based in part on the previously mentioned study. A brief overview of this "draft payload requirements document" is also provided below.

A current effort to establish a NASA-wide software safety analysis and management standard is underway. This work is documented in the draft "Software Safety Standard," dated May 26, 1993. An overview is provided below.

Two relevant systems-related NASA documents that apply to the Space Station Freedom Program (SSFP) are as follows:

- SSP 30000, "Space Station Program Definition and Requirements," and in particular, Section 9 on Product Assurance Requirements and Section 12 on Space Station Program Master Verification Requirements, and

- SSP 30309 (Rev. B, 1991), "Safety Analysis and Risk Assessment Requirements Document."

Section 12 (and recent changes thereto, e.g., Change 2 performed in 1992) of the former document specifies the verification requirements for the Space Station Freedom Program. Verification in this context is, in general, the process of ensuring that requirements are met relative to functions, performance, operations, quality and safety. A related document (currently in draft form) entitled "Program Master Verification Plan: Avionics and Flight Software Integration and Verification Plan" (TSS 30666, Volume 4, Part 1), provides an overview of the integration and verification processes for avionics and flight software at the program level (i.e., Space Station Freedom Program). An overview of the second item listed above (i.e., SSP 30309) is provided below.

Relevant documents prepared for NASA by the Jet Propulsion Laboratory (JPL) in Pasadena, California, are as follows:

- JPL D-576, "Independent Verification and Validation of Computer Software: Methodology," 1983, and

- JPL D-10058, "Software Systems Safety Handbook," May 1993.

The first document describes an approach to perform software IV&V, and is based upon the identification and description of IV&V tasks and tools that are integrated into the development life cycle. The latter document, directed primarily to software developers, describes the results of a recent study by JPL of software safety issues. To date, it has not yet been established as a NASA-wide Handbook.

There is also considerable work being conducted by NASA and their contractors in the area of formal methods, which involve the application of applied mathematics to computer system engineering. These can involve the development of formal specifications and the use of formal proof techniques. NASA Langley is heavily involved in the design and formal verification of a fault-tolerant computing platform that would be suitable for advanced flight control applications.

3.24.1 NHB 1700.1 (V1-B)

Chapter 3 of this document (NASA Safety Policy and Requirements Document) describes NASA-wide policies and requirements for the establishment and implementation of system safety processes to identify and reduce safety risks to acceptable levels. A Safety Management Plan is to be prepared that provides the basis for all activities to be conducted. Safety analyses are to performed throughout the system life cycle to systematically identify hazards, determine their risk level and provide means for their elimination/control. Analyses to be performed include the following: Preliminary Hazard Analysis (PHA), Subsystem Hazard Analysis (SSHA), Software Hazard Analysis (SWHA), System Hazard Analysis (SHA), Operating and Support Hazard Analysis (O&HA), and Integration Hazard Analysis (IHA). These analyses are to use data from other analyses such as FMEAs, Critical Items Lists, Operations Analysis, Human Engineering Analysis and Maintainability Analysis. Appendix H of the document provides additional information on the specific analysis types.

Program and change reviews are other tasks that are to be performed.

3.24.2 Draft Payload Requirements Document

This draft document describes high-level requirements for the design and V&V of computer-based systems used in potentially hazardous payload applications for the Space Shuttle program. Requirements include the use of a formal development process (including peer reviews) and a security system for software, a safety analysis of potentially hazardous hardware and software, a stress analysis on hardware, qualification of hardware and software including an IV&V effort for the software, and acceptance and validation testing. Hardware and software design requirements are described. It should be emphasized that this draft represents some early efforts to create top level requirements for payload safety.

3.24.3 SSP 30309 Rev B

This document presents requirements for safety analyses and risk assessments within the Space Station Freedom Program, including those directed to computer-based systems. Paragraph 3.0 describes the hazard analysis process to be used, including the identification, evaluation and classification of the hazards. The minimum system/hardware related analyses to be performed include the following: PHA, SSHA/SHA, SSHA, O&SHA and FMEA cross-checks. Software analyses are to be an integral part of many of the above analyses (i.e., PHA, SSHA/SHA, O&SHA). Specific software analysis techniques to be used are discussed in Appendix D of the document, and include such techniques as: Software Requirements Analysis, Criticality Analysis, Specification Analysis, Timing and Sizing Analysis, Design Logic Analysis, Design Data Analysis, Design Interface and Constraint Analyses, as well as Code Data, Interface and Constraint Analyses. In addition, a fault tree analysis is to be performed to supplement the identification of hazards.

Software testing is also to be used to verify analysis results, investigate program behavior as needed and confirm compliance with safety requirements. It is to include nominal, stress and performance testing in either a controlled or demonstration environment.

It appears that Revision C of this document exists, but it was not available for this report.

3.24.4 Draft Software Safety Standard

NASA's Draft Software Safety Standard (current version dated May 26, 1993) describes the activities and organization considered necessary to ensure that safety is designed (in a cost effective manner) into NASA-developed software. It is being developed within the Office of Safety and Mission Assurance at NASA Headquarters. The intent is for this standard to be used NASA-wide in conjunction with NHB 1700.1. The primary reference documents for this draft standard are DOD-STD-2167A and MIL-STD-882B, Notice 1.

The focus of this document is a software safety analysis that is to be performed as part of the overall system safety effort. Specific tasks (of which there are seven) within the analysis process are to be integrated into the software development process. The general nature of these tasks are discussed below.

3.24.4.1 Software Requirements Analysis (SRA) - In this analysis, the system/software requirements are to be examined for possible unsafe modes. Results of a system PHA are to be used as key input here. This task is to include establishing and implementing a requirements-tracking system, analyzing software requirements specifications, and developing recommendations and testing requirements.

3.24.4.2 Software Top-Level Design Analysis - This analysis is to include the identification and analysis of the Safety Critical Software Components (SCCSCs), the recommendation of changes as appropriate, and integration of safety requirements into the Software Test Plan.

3.24.4.3 Detailed Design Analysis - This analysis is to verify the correct implementation of the detailed design (prior to coding) using results of the previous two analyses. This is to involve a hazard risk assessment. Changes are to be recommended as necessary, and inputs provided to the Software Test Plan. Safety critical software units are to be identified for the code developers.

3.24.4.4 Code Analysis - Program code and system interfaces are to be examined for their impact upon safety. This is to include the conduct of a Process Flow Analysis.

3.24.4.5 Software Safety Testing - Software is to be tested to ensure that all hazards have been eliminated or controlled to an acceptable level. This is to be performed within the software's specified environment as well as in abnormal conditions.

3.24.4.6 Software Interface Analysis - This analysis is to identify those hazards that are not eliminated or acceptably controlled by the design, and to recommend appropriate detection, warning, or annunciation provisions.

3.24.4.7 Software Change Analysis - All changes made to the software should be evaluated for their impact upon safety and analyzed as necessary.

3.25 NUCLEAR REGULATORY COMMISSION

The Nuclear Regulatory Commission (NRC), headquartered in Washington, D.C., has established general requirements for the design, verification and validation of safety-related equipment/systems in nuclear power plants. These requirements are found in the following portions of 10 CFR Part 50:

- Appendix A, Criterion 21 (Protection System Reliability and Testability) -- requires that protection systems be designed for high functional reliability in accordance with the safety functions to be performed, and

- Appendix B, Criterion III (Design Control) -- requires that quality standards be specified and design control measures be provided for verifying/checking the adequacy of the design.

The primary document that provides guidance on complying with these regulations (for computer-based systems) and establishes the NRC's current position is Regulatory Guide 1.152, "Criteria for Programmable Digital Computer System Software in Safety-Related Systems of Nuclear Power Plants," dated November 1985. This guide endorses the following document jointly prepared by the American Nuclear Society (ANS) and the Institute of Electrical and Electronics Engineers (IEEE):

- ANSI/IEEE-ANS-7-4.3.2-1982, "Application Criteria for Programmable Digital Computer Systems in Safety Systems of Nuclear Power Generating Stations."

This document describes a general method for designing/implementing software and validating computer systems used in safety-related systems of nuclear power plants. It is intended to provide a common understanding on, among other things, software verification and validation procedures for the nuclear industry. It was developed to supplement the IEEE Standard 603-1980 ("Standard Criteria for Safety Systems for Nuclear Power Generating Stations") by incorporating information/criteria relative to computer-based systems. A brief overview of this existing document (i.e., ANSI/IEEE-ANS-7-4.3.2-1982) is provided below. As will be seen, it references some earlier quality assurance requirements (NQA-1-1979) produced by the American Society of Mechanical Engineers (ASME).

Formal certification of computer-based systems or software is not required by the NRC. However, a safety review/evaluation (or audit) is typically conducted by Nuclear Reactor Regulation (NRR) office staff on a vendor's system to ensure compliance with requirements, including those pertaining to system/software verification and validation.

Recent efforts by a joint ANS/IEEE working group are being directed to revising the 1982 ANSI/IEEE-ANS-7-4.3.2 standard. At this time the revision is still in draft form, and is identified as follows:

- P-7.4.3.2, Draft 7, "Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations."

Its purpose is provide additional requirements (including verification and validation) for computer-based systems by updating the earlier 1982 version and supplementing the criteria and requirements in IEEE Standard 603-1991. It also will describe the relationship between other relevant documentation. In particular, two key documents dealing with quality assurance (including verification and validation) will be referenced by the new P-7-4.3.2 standard. Those are:

- ASME NQA-1-1989, "Quality Assurance Program Requirements for Nuclear Facilities," and

- ASME NQA-2a-1990, Part 2.7, "Quality Assurance Requirements of Computer Software for Nuclear Facility Applications."

An overview of this new draft is also provided below, following a discussion of the existing standard.

It should also be noted here that Ontario Hydro and the Atomic Energy Commission Board (AECB), in Ottawa, Canada, are heavily involved in establishing and implementing safety standards for nuclear power plants, especially those pertaining to software safety, software engineering, quality assurance and related topics. One recent effort resulted in a document entitled "Standard for Software Engineering of Safety Critical Software," (982C-H69002-0001). This document defines requirements (including the area of V&V) for software engineering of real-time control systems in nuclear generating stations. It draws heavily upon the content of the following two documents:

- IEC Std Pub 880, "Software for Computers in the Safety Systems of Nuclear Power Stations" (1986), and

- CAN/CSA-Q396.1.1-89, "Quality Assurance Program for the Development of Software Used in Critical Applications."

The first document is currently being updated and will address such topics as qualification of pre-existing software, formal methods, CASE tools and software design diversity. The latter is one of several quality assurance documents issued by the Canadian Standards Institute.

Another key document utilized by Ontario Hydro, and one that addresses computer hardware safety, is IEC 987, "Programmed Digital Computers Important to Safety for Nuclear Power Stations." This document defines requirements in a large number of areas for computer hardware including design, verification, validation, documentation and others.

3.25.1 ANSI/IEEE-ANS-7-4.3.2-1982

This existing standard (consisting of 11 pages) presents application criteria for computer systems relative to the design and evaluation of safety performance and reliability. Evaluation in this context refers to the verification and validation processes which are considered essential throughout the development of computer-based systems. As stated earlier, it was created to supplement IEEE Std. 603-1980, and in particular, to address issues relating to computer-based systems (i.e., software design, software implementation and computer system validation).

An overall system development process is presented. First, computer system design and documentation requirements are generated for hardware, software and hardware-software integration aspects. These are based upon overall safety system requirements, and should include acceptance criteria for validation. Areas for which requirements should be established are listed. Then, hardware and software are to be developed to meet the specific requirements. Hardware is to be developed according to IEEE Std. 603-1980.

Software development is to involve three phases: development plan, design and implementation. The development plan is to specify the standards and procedures to be used in the development, including appropriate quality assurance provisions for the software such as those obtained from the following:

- ANSI/ASME NQA-1-1979, "Quality Assurance Program Requirements for Nuclear Power Plants," or

- IEEE STD 467-1980, "Standard Quality Assurance Program Requirements for the Design and Manufacture of Class 1E Instrumentation and Electric Equipment for Nuclear Power Generating Stations."

Hardware-software integration takes place after the hardware and software design phases.

3.25.1.1 Verification - Verification in this document involves determining the correct transition from each stage in the development process to the next. Its intent is to minimize design errors in the system. Minimum requirements are presented in three areas: organization, review/audit procedures, and software test/analysis. In the area of organization, it is suggested that the verification group be independent from the design team. For reviews/ audits, it is suggested that detailed procedures and policies be established and implemented to review and audit design documentation, specifications and plans. In addition, procedures and practices for software testing and analysis should be established and implemented to supplement reviews and audits.

3.25.1.2 Validation - Validation is suggested for the hardware and software as a system, following ali development and verification activities. The computer system is to be exercised through both static and dynamic simulation of input signals to confirm proper normal and design event conditions of operation. A formal test plan describing required inputs, expected outputs and acceptance criteria should be generated. This testing is to be done by an independent team from the design/implementation group. Guidance is also provided for documentation of the results.

3.25.2 P-7-4.3.2 Draft 7

This draft document, "Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations," represents the current effort (at the time of this report) of a joint ANS/IEEE working group to create an updated standard for computer-based systems in safety-related nuclear power plant applications. It should be noted that this current draft contains over 55 pages — compared to the 11 pages of the earlier version. Its intent is to present guidance on design requirements for computer systems and methodologies/techniques for demonstrating compliance with those requirements. As with the earlier version (discussed above), it is meant to supplement the updated IEEE Std. 603-1991.

A significant number of safety criteria in a wide variety of areas are listed and discussed. One of these areas is referred to as "quality," which encompasses the following six subtopics: software development, manufacturer's qualification of existing commercial product, software tools, verification & validation (V&V) and configuration management. In this document, V&V is defined as it is in the IEEE Std 610.12, and as follows:

- V&V — process of determining whether the requirements for a system or component are complete and correct, the products of each development phase fulfill the requirements and conditions imposed by the previous phase, and the final system or component complies with specified requirements.

Section 5.3.5 of this draft as well as Appendices E and F deal with the topic of V&V. Essentially, V&V is to address computer software and hardware as well as non-computer hardware throughout development, and is to include system testing of the final integrated hardware, software, firmware and interfaces. A V&V plan is to be prepared in crder to document all activities to be performed. Guidance for this plan may be obtained in one or both of the following two documents:

- ANSI/IEEE Std 1012-1986, "IEEE Standard for Software Verification and Validation Plans" (described in more detail in the IEEE section of this report), and/or

- IEC Std Pub 880, 1986, "Software for Computers in the Safety Systems of Nuclear Power Stations."

Hardware verification is to be performed in accordance with ASME NQA-1-1989. In general, software V&V is to be performed in accordance with sections 3 and 4 of ASME NQA-2a-1990, Part 2.7. According to P-7-4.3.2, quality assurance is applied to software development in order to develop reliable, high quality software and to minimize errors. V&V is to ensure that the system design, including the handling of all "credible abnormal conditions," has been properly (and safely) implemented. Emphasis in this overview has been given to the V&V process and related matters described in Appendix E (Verification and Validation) and Appendix F (Identification and Resolution of Abnormal Conditions and Events).

3.25.2.1 V&V Process Summary - According to this draft standard, V&V is performed to ensure the (correct) implementation of requirements and minimize the potential for deficiencies that could result from the software development process. It is to involve activities which are performed throughout the overall computer system and software development processes. These activities are to be independent from the design activities (as directed by NQA-1-1989), and as such, are to be performed by persons external to the design team.

V&V activities are to include a combination of reviews, inspections, analyses and testing. Analytical based methods can include the following:

- Independent reviews -- ensures the traceability (via traceability matrix) of safety design basis requirements

- Independent witnessing -- witness designer activities

- Inspection -- walkthroughs of design, code and documented test results

- Analysis -- options include formal proofs, petri nets and other graphical analysis methods; formal methods are recommended for small sections of code with well defined functions.

Testing can be used on any aspect that is executable or compilable such as a rapid prototype, executable specification, program design language, hardware and software interface, or applications code. It should also be used to help ensure that non-safety functions do not adversely impact safety functions. The two basic types of testing recommended here are summarized below:

- Functional testing -- ensures functional behavior is consistent with requirements; includes black box testing, factory acceptance testing and site acceptance testing; addresses only inputs and outputs of components under test; effective for module's interface and fault detection/containment aspects of interface

- Structural (white box or glass box) testing -- addresses internal structure of code; accomplished via branch or path testing; in branch testing, test cases should ensure all branches are traversed; in path testing, test cases should ensure all feasible combinations of branches are traversed.

Appendix E of the document presents an overall computer development process, and indicates V&V activities that should be associated with each activity of the process. Two major development approaches are discussed: sequential and iterative. In the sequential approach, each phase is completed in an ordered manner. In the iterative approach (sometimes referred to as spiral development methodology), requirements, designs and implementations are changed multiple times throughout development.

In addition to normal operation, V&V activities are to address abnormal conditions and events (ACEs). This topic is treated in more detail later.

The various V&V activities to be performed are briefly summarized below:

- Requirements allocation V&V -- ensures appropriate allocation of non-computer hardware, computer hardware, and software requirements via reviews or inspections; also, an FMEA should be performed at this point to identify failure modes, design basis events and hazards -- this will help in the definition of lower level requirements; FMEA guidance can be obtained via ANSI/IEEE Std 352-1987, "IEEE Guide for General Principles of Reliability Analysis of Nuclear Power Generating Station Safety Systems"

- Non-computer hardware requirements V&V -- ensures requirements meet IEEE Std 603-1991

- Computer hardware requirements V&V -- ensures integrity, single failure and independence criteria are met

- Software requirements V&V -- review of Software Requirements Specification to ensure computer requirements for software are met; ensure software addresses fault detection, handling and containment, and diagnostics are addressed; traceability matrix is suggested; system and acceptance test plans should be initiated

- Integration requirements V&V – review of computer hardware and software integration documentation to ensure adequacy of interfaces: could include integration plan, test procedures and acceptance criteria

- Non-computer hardware design and implementation V&V – ensures design addresses non-computer hardware requirements including interfaces with computer hardware and software

- Computer hardware design and implementation V&V – ensures design addresses computer hardware requirements including interfaces with other hardware and software

- Software design and implementation V&V – to be performed in two parts as follows:

  - Software design V&V – based upon inspections or reviews: ensures software requirements are addressed in a Software Design Description document (such as recommended in IEEE Std 1016-1987; also ensures that requirements imposed upon software by hardware is met; identify commercial grade software items (as they are to be treated differently).
  - Software implementation V&V – typically based upon testing and analysis; ensures that software design has been correctly implemented; aspects to be considered include the following: algorithm analysis, database analysis, control flow analysis, and sizing and timing analysis. Note: NQA-2a-1990, Part 2.7, Sections 3, 4 and 7, provide additional requirements on software V&V for nuclear facility applications. It appears that efforts are underway to revise/consolidate software V&V requirements. These may then be incorporated into or referenced by the final version of the P-7-4.3.2 standard.

- Computer integration V&V – to be performed on integrated system elements; can assess conformance with response time, throughput, interface and functional requirements; can assess conformance to fault handling/containment requirements via fault simulation/injection methods; regression testing should be performed as appropriate

- System (factory acceptance) testing V&V – ensures the correct implementation of requirements for the computer; additional guidance can be found in Section 8 (Computer System Validation) of IEC Std Pub 880, 1986

- Site Acceptance V&V – performed in nuclear power generating station; ensures the correct implementation of safety requirements in application environment

- V&V documentation — to be prepared in accordance with IEEE Std 1012-1986; final report should describe compliance with V&V plan.

3.25.2.1.1 Abnormal Conditions and Events - As mentioned earlier, one key aspect of V&V which, according to this draft standard, must be considered throughout the computer development process involves the identification of abnormal events and conditions (ACE). Appendix F of this draft document (P-7-4.3.2) provides guidance on identifying and resolving ACEs. The recommended activities to address the ACE issue are based upon the methods of two main documents:

- IEEE Std P1228, "Software Safety Plans" (currently in draft stage and addressed in the IEEE section of this project report), and

- MIL-STD-882B, "System Safety Program Requirements" (addressed in the DOD section of this report).

Examples of ACEs include failure modes of system components, common mode/cause failures, human error, interface incompatibility, improper input/output timing, and out-of-sequence events. Appendix F provides a detailed discussion on identifying ACEs at various design stages (e.g., requirements, design, integration). In general, ACEs can be identified by a combination of fault tree and FMEA techniques. As described earlier, ACEs should serve as input to the V&V process as appropriate.

Guidelines for resolving ACEs are based upon one of three general techniques (in descending order of priority): elimination of the ACE, use of warning devices, or use of appropriate procedures/training.

3.25.2.2 Modifications - P-7-4.3.2 points to Section 9 (Maintenance and Modification) of IEC Std Pub 880, 1986, for guidance in addressing modifications to (essentially) the software. In general, regression testing should be used to ensure changes do not adversely impact the system. ACEs introduced as part of the changes should also be addressed as described in Appendix F of the document. There is further guidance on modifications in NQA-2a-1990, Part 2.7, Section 4.2 (Software Validation) — this also suggests regression testing.

3.25.2.3 Other Issues - Several other issues relating to system safety are addressed by this draft document. Included are diversity, EMI, use of commercial grade components (items not developed under this standard), data communications and quantitative/software reliability. Design and, in some cases, validation issues are addressed. Regarding the latter, the document suggests that existing methods of predicting software reliability do not provide adequate confidence in their results. Therefore, a general method for making a software reliability measurement (with some limitations) has been developed and is presented.

3.26 MEDICAL INDUSTRY

Several initiatives are underway in the medical industry both nationally and internationally to address safety concerns pertaining to the use of computers in medical applications.

One such effort in the U.S. is being headed up by the Food and Drug Administration (FDA), who has the responsibility of assuring both the safety and effectiveness of medical devices. To this end, the FDA is preparing a policy (i.e., "FDA Policy for the Regulation of Computer Products," draft, 1989) on how to determine whether a computer product is a medical device and how it is to be regulated.

In addition, the FDA's Center for Devices and Radiological Health has issued the document "Reviewer Guidance for Computer Controlled Medical Devices Undergoing 510 (K) Review," which primarily addresses software aspects of medical devices. This document, summarized in more detail below, provides guidance to an FDA reviewer of "premarket notification" or 510 (K) submissions for medical devices. In general, it describes 1) what the FDA reviewer is to look for in the development and safety assurance activities/documentation (including V&V) performed/generated by the manufacturer, and 2) guidance on the review process itself.

Another effort underway by the FDA is the establishment of requirements for a manufacturing process pertaining to computer-based medical equipment. These requirements are documented in a draft report (dated November 1990) entitled "Application of the Medical Device GMPS (Good Manufacturing Practices) to Computerized Devices and Manufacturing Processes-Medical Device GMP Guidance for FDA Investigators."

An international effort currently underway by the International Electrotechnical Commission (IEC) involves the development of a standard for the development and safety assessment of computer-based medical equipment. This work is currently documented in the draft IEC 62 (Secretariat) 69, "Electrical Equipment in Medical Practice," dated March 1993. It is intended to address computer systems and supplement the existing IEC 601-1 series of standards which address various safety aspects (e.g., testing, EMC) of (primarily) non-computer-based medical devices. An overview of this draft standard is also provided below.

3.26.1 <u>Reviewer Guidance for Computer-Controlled Medical Devices Undergoing 510 (K) Review</u>

As indicated above, this document provides guidance on the kind of information (including software development and safety V&V) FDA reviewers (should) see in 510 (K) submissions and the approach to be used in reviewing the computer-based devices/information. It emphasizes the importance of the software development process (including analyses and reviews) in addition to the testing of the developed product.

Following a discussion of "levels of concern" for classifying the safety criticality of the devices (which help determine the depth of review needed), the following three areas are addressed: development/documentation of software, system documentation and review task.

3.26.1.1 Development/Documentation of Software - Five phases of software development are expected of manufacturers: specifications/requirements, design, implementation, verification and validation, and maintenance. V&V activities are to include testing and other methods at the unit, integration and system level as well as assessing the quality of the software throughout the development process. The conduct of hazard analyses and demonstration that hazards have been eliminated/minimized is expected. Other techniques that may be expected from the manufacturer include requirements analysis, design reviews, code walkthroughs, control flow analysis, software complexity metrics, data flow analysis, fault tree analysis and FMECAs. In more complex systems, testing of software is expected at the module, function and subroutine levels. All safety-related functions are to be exercised in the tests.

3.26.1.2 System Documentation - System/software documentation is to include descriptive information and results relevant to areas such as functional requirements and specifications, software design/development, software V&V and testing.

3.26.1.3 Review Task - The reviewer's activities in the area of V&V is to include confirmation of the following (and numerous other) items:

- Hazards have been identified and accounted for in the design
- Software development process includes quality assurance activities plus V&V and other testing methods/procedures
- Results of V&V, testing, etc. demonstrate that the device is safe.

The remaining portion of the document presents detailed questions that could be asked/considered by the reviewer. These are listed by three different levels of concern for the device.

3.26.2 IEC 62 (Secretariat) 69

This draft international standard, "Electrical Equipment in Medical Practice," describes general safety requirements for designing and assessing computer-based medical equipment. It stresses the importance of the development process in the overall safety assurance scheme, and presents an approach to identify hazards and manage associated risks. Brief summaries of the requirements in seven major activities are provided below:

- Hazard Identification — list all hazards with emphasis on the patient as opposed to the equipment; hazards are to include those which could occur in normal use, under failure conditions, in association with the material environment, and in association with the operator/user
- Security Level Assignment — rank safety hazards according to severity levels

- Risk Requirements Establishment – establish likelihood of hazards and determine risk requirements
- Selection/Development of Design Options – determine design options for managing risks
- Design Option Analysis – evaluate design options
- Risk Control – select design options and determine final risk requirements
- Risk Management Record – documentation of risk management process.

Specific hazard analysis and software assurance/safety techniques are not identified, but some suggested techniques include those found in such documents as: MIL-STD-1629A (FMECA), MIL-STD-882B, ISO 9000-3 (quality management/assurance), IEC 65A (Secretariat) 122 and 123, DIN V VDE 0801, "Reviewer Guidance for Computer Controlled Equipment" (FDA) and others.

3.27 UNDERWRITERS LABORATORY

Underwriters Laboratory (UL), Inc., in Northbrook, Illinois, has developed a safety standard (i.e., UL 1998) for software due to the increasing usage of computers in the implementations of safety critical functions in consumer products. A draft of this standard, namely, "Proposed First Edition of the Standard for Safety-Related Software" (dated July 30, 1993), was available at the time of this report. This standard is intended to be used in conjunction with end-product and hardware standards to conduct an overall investigation of a computer-based system. In fact, the UL 1998 standard requires an integral investigation of the controlling hardware according to UL 991, "Standard for Tests for Safety-Related Controls Employing Solid-State Devices."

A brief overview of the UL 1998 July 30th draft for software is provided below. It should be noted that the (final) first edition of UL 1998 was published in January 1994, but was not available in time to be included in this report. Therefore, there could be slight differences between the draft discussed below and the final version.

3.27.1 UL 1998 (Draft) Software Standard

The requirements in UL 1998 pertain primarily to the software development process, and are directed to three main aspects: risk analysis, design, and code level analysis/testing. However, the requirements also briefly address a number of other topics such as documentation, configuration management, and software modifications.

3.27.1.1 Risk Analysis Requirements - The standard requires a risk analysis to be performed in order to determine that the software addresses the possible risks as intended and does not introduce any new risk. The two methods to be used are failure modes and effects analysis (FMEA) and fault tree analysis (FTA). The FMEA is to be used to identify "critical" and "non-critical" portions of the software, and the FTA is to be conducted to identify conditions that could result in an (unacceptable) risk.

3.27.1.2 Design Requirements - Certain general and more specific software design requirements (for critical and "supervisory" software) are identified such as those which pertain to software modularity, accessibility of memory regions, critical data use and numerous others. There are also requirements which pertain to outputs, as well as system issues such as power outages and user interfaces.

The design requirements given the most attention in the standard are those which pertain to measures that can be used to address (or protect against) hardware failures and malfunctions. Measures that can be used to protect against certain types of hardware failures are identified for two different software classes: Software Class 1 and Software Class 2. Software Class 2 is associated with "special risks" such as explosions, and generally has more stringent design requirements assigned to it.

3.27.1.3 Code Level Analysis and Test - The following analysis and test requirements are identified for the software development process:

- Program logic and data analysis – to conduct an analysis of program logic and data to determine that software addresses possible risks and does not introduce new risk; includes evaluation of each decision criterion and function that could involve a risk; test cases are to be developed

- Code level analysis – to determine that software only performs intended functions and does not result in a risk; includes analysis of critical software portions for completeness and correctness, and that implementation complies with end-product requirements; includes analysis of outputs and response to inputs; includes analysis of source code for possible combinations of hardware failures, software errors, transient errors and other events; includes analysis of shut-down procedures

- Operational test – to determine compliance of critical software portions with requirements of the standard via testing; includes development of test plan, test parameters, test procedures, test criteria, and test cases

- Failure mode test – to determine that software responds correctly to single failures (e.g., operator errors, component failures), and

- Software partition analysis – to verify the integrity of software partitioning.

4. SAFETY VERIFICATION AND VALIDATION METHODOLOGY ASSESSMENT

This section presents the results of the assessment on the safety verification and validation methodologies. As discussed earlier, the assessment was conducted in two parts: initial assessment and detailed assessment. Results of the initial assessment are presented first, followed by results of the detailed assessment which includes the identification of attributes and limitations of the various methodologies addressed.

4.1 INITIAL ASSESSMENT

The purpose of this first assessment was to conduct an initial screening of the existing methodologies in order to select the most appropriate candidates for further review.

4.1.1 Initial Assessment Criteria

Criteria utilized for this assessment were based upon some general aspects of the methodologies as well as the potential applicability of the methodologies to railroad and other fixed guideway equipment. The four criteria utilized are listed below, followed by brief discussions on each:

- Standards involved in the safety V&V process
- Detail available on the process/activities utilized/developed
- General applicability to railroad and other fixed guideway equipment
- General applicability to different configurations/design philosophies.

4.1.1.1 Standards Involved in the Safety V&V Process - Probably the most significant criterion in this portion of the assessment pertains to the involvement of existing national, international or industry standards/guidelines in the methodologies utilized or developed by the various organizations. In particular, it has been observed that some of the safety V&V methodologies utilized by certain organizations are based entirely upon an existing standard, while others are based upon multiple standards or portions thereof. Still other firms have their own methodologies which may take into account the intent of existing standards or may be a totally unique process (particularly where national standards do not exist). There are, of course, other firms that have developed standards or guidelines for utilization by others.

The primary purpose of applying this criterion was to summarize and tabulate the key standards or major guideline documents involved in the safety V&V methodologies of the organizations of interest.

This criterion has particular significance in that the overall objective of the Base Task in this study is to identify and recommend the "best" existing methodology which could serve as a basis for the development of a refined and improved methodology for FRA's consideration. The accomplishment of this objective is best facilitated if the recommended methodology (from the Base Task) is based upon an existing standard or, perhaps, other well-documented process.

4.1.1.2 Detail Available On Process/Activities Utilized - The purpose of this criterion was to assess the availability of detailed information on the safety V&V methodologies utilized or developed by the various firms. If one or more associated and relevant standards were identified for an organization and obtained for this study, the necessary detail was considered available. However, if a standard was not identified, it was necessary to determine whether or not sufficient detail was otherwise available on the V&V activities performed to allow a proper assessment/comparison of the methodology to be made. It is recognized that even though numerous relevant standards exist, they address and describe safety V&V processes and activities in varying levels of detail.

4.1.1.3 General Applicability to Railroad and Other Equipment - The purpose of this criterion was to assess the general applicability of the safety V&V methodology to railroad and other fixed guideway equipment. It was not the intent in this initial assessment to determine the comprehensiveness or effectiveness of the methodology or how well it could be applied, but rather, whether or not it could be applied at all to the equipment of interest. A more thorough investigation as to its applicability is addressed in the detailed assessment. The detailed assessment also addresses how well the methodology could be applied to other technology (e.g., relay-based or hardware-only systems) in addition to computer-based systems, which are the primary interest in this study.

Equipment of particular interest in the railroad industry includes safety critical computer-based signalling/train control, communications and other systems (e.g., grade crossing systems) that could involve wayside, on-board and even centrally located equipment in conventional as well as high-speed rail applications. Examples include interlockings, track circuits and other train detection equipment, speed measurement/control systems affecting propulsion and/or braking and enforcing safe speed limits, and data communications equipment responsible for transmitting, receiving, encoding and decoding safety critical data. Other safety critical computer-based equipment of interest involves that used in other fixed guideway applications such as in maglev. Examples here include equipment pertaining to the control of vehicle speed and guideway power, vehicle separation, levitation, guidance, switching and safety related communications.

4.1.1.4 General Applicability to Different Design Philosophies - One other key aspect of the initial assessment involved the determination as to whether the methodology could be applied to different computer system design philosophies including different hardware and software configurations. Again, the intent here was not to determine the effectiveness of a given methodology on different philosophies or how well it could be applied, but rather, whether it could generally be applied, in part or in full, to a wide variety of systems/equipment.

This criterion is important in that a number of different design philosophies and configurations for safety critical computer-based systems are being utilized in North America as well as overseas, and a methodology is (currently) needed to address the different philosophies/configurations that may be experienced in the U.S. Limiting the methodology to a single design philosophy or configuration would, of course, greatly limit its usefulness, unless a decision is later made to require a specific configuration.

The following design philosophies/configurations are currently being used in computer-based safety critical systems:

- Single channel systems (essentially one microprocessor performing any given function in a single data path) with extensive embedded diagnostics

- Single channel systems based upon special embedded software coding and signature techniques

- Single channel systems with multiple/diverse software programs

- Dual channel redundant (hardware) systems with hardware and/or software comparators

- Triple channel systems with voting schemes.

4.1.2 Initial Assessment Summary

Results of the initial assessment are presented in Table 4-1. The organizations in the leftmost column are generally listed in the same order in which they were addressed in Section 3 of this report. It should be noted that several other organizations are shown in the table in addition to those specifically addressed and/or discussed in Section 3. This is because some additional relevant standards were identified since the detailed methodology descriptions were completed.

TABLE 4-1. INITIAL SAFETY VERIFICATION/VALIDATION ASSESSMENT

Organization	Initial Assessment Criteria				
	Standard(s) Involved In Safety V&V Process	Detail Available on Process/Activities	Applicable to RR & Other Fixed Guideway Applications	Applicable to Different Configurations/ Design Philosophies	Comments
GRS	None (Internal guidelines only)	Limited	Yes	No	Process geared to company design philosophy
US&S	None (Internal guidelines only)	Limited	Yes	No	Process geared to company design philosophy
Harmon	None (Internal guidelines only)	Limited	Yes	No	Process geared to company design philosophy
ALCATEL	None (Internal guidelines only)	No	Yes	No	Process geared to company design philosophy
Railway Association of Canada (Advanced Train Control System - ATCS)	ATCS Spec 140*	Yes	Yes	Yes	Primary safety document
	ATCS Spec 130*	Yes	Yes	Yes	Addresses software quality assurance only
British Rail	RIA Tech Spec No. 23* (plus other internal standards)	Yes	Yes	Yes	Internal standards are proprietary
International Union of Railways (UIC)	UIC 738 R*	Yes	Yes	Yes	General design/assessment guidelines only
	UIC/ORE A155/RP11*	Yes	Yes	Yes	Describes system, H/W and S/W validations
	UIC/ORE A155.1/RP8*	Yes	Yes	Yes	Addresses data transmission systems

* Selected for Detailed Assessment

TABLE 4-1. INITIAL SAFETY VERIFICATION/VALIDATION ASSESSMENT (cont.)

Organization	Initial Assessment Criteria				
	Standard(s) Involved In Safety V&V Process	Detail Available on Process/Activities	Applicable to RR & Other Fixed Guideway Applications	Applicable to Different Configurations/ Design Philosophies	Comments
TÜV Rheinland	No formal standard for TÜV's process—SBT 90.01/00/E is informational document only	Limited—s/w only	Yes	Yes	Safety V&V activities are project dependent
German National Standards Institute (DIN)	DIN V VDE 0801*	Yes	Yes	Yes	Primary German standard (except for DB)
	DIN V 19250*	Yes	Yes	Yes	Risk assessment only
	DIN VDE 0831*	Yes	Yes	Yes	Installation oriented
German Federal Railway (DB)	Mü 8004*	Yes	Yes	Yes	Proof-of-safety requirements (extensive)
CENELEC (European)	CLC/TC9X/SC9XA/WGA1* (Late Draft)	Yes	Yes	Limited	Addresses system and H/W aspects mainly
	CLC/TC9X/SC9XA/WGA2 (Early Draft)	No	Yes	Limited	Addresses S/W; draft to be available in early 1994
ABB Signal AB	None, other than proprietary internal standards	Limited	Yes	No	Process geared to company design philosophy

* Selected for Detailed Assessment

TABLE 4-1. INITIAL SAFETY VERIFICATION/VALIDATION ASSESSMENT (cont.)

Organization	Initial Assessment Criteria				
	Standard(s) Involved In Safety V&V Process	Detail Available on Process/Activities	Applicable to RR & Other Fixed Guideway Applications	Applicable to Different Configurations/ Design Philosophies	Comments
Siemens AG	DIN V VDE 0801*	Yes	Yes	Yes	
	DIN V 19250*	Yes	Yes	Yes	
	Mü 8004*	Yes	Yes	Yes	Primary document for DB applications
	Plus two key internal standards	Limited	Yes	Unknown	For S/W development and assessment
Matra Transport	None specifically cited, but process used may cover intent of French standards NF F 71-011, NF F 71-012, NF F 71-013	Limited	Yes	Yes	Extensive safety V&V process with formal methods for S/W; French standards not currently available in English
Sasib	Internal guidelines	Limited	Yes	Unknown	Italian State Railway Standard IS 402 also applies, but is not directed to safety V&V
French National Railway (SNCF)	Information not available/ provided	No	-	-	Process may involve French standards NF F 71-011 thru -013 plus internal standards (e.g., DEI 100)

* Selected for Detailed Assessment

TABLE 4-1. INITIAL SAFETY VERIFICATION/VALIDATION ASSESSMENT (cont.)

Organization	Initial Assessment Criteria				
	Standard(s) Involved In Safety V&V Process	Detail Available on Process/Activities	Applicable to RR & Other Fixed Guideway Applications	Applicable to Different Configurations/ Design Philosophies	Comments
Ministry of Defense (MOD)	Interim Defense Standard MOD 00-55*	Yes	Yes	Yes	S/W development
	Interim Defense Standard MOD 00-56*	Yes	Yes	Yes	Risk assessment and hazard analysis
Institute of Railway Signal Engineers (IRSE)	No formal standard, but recommendations provided in IRSE Report No. 1* (Cross Acceptance)	Yes	Yes	Limited	Recommends use of international standard such as IEC 65A
Railway Technical Research Institute (RTRI)	None identified	Limited	Yes	Unknown	Process used similar in nature to MIL-STD-882
East Japan Railways	Information not available/ provided	No	-	-	
Nippon Signal	None identified	Very limited	Yes	Unknown	Safety V&V process currently being revised

* Selected for Detailed Assessment

TABLE 4-1. INITIAL SAFETY VERIFICATION/VALIDATION ASSESSMENT (cont.)

Organization	Initial Assessment Criteria				
	Standard(s) Involved In Safety V&V Process	Detail Available on Process/Activities	Applicable to RR & Other Fixed Guideway Applications	Applicable to Different Configurations/ Design Philosophies	Comments
International Electrotechnical Commission (IEC)	IEC 65A (Sec.) 122*	Yes	Yes	Limited	S/W development; still in draft form
	IEC 65A (Sec.) 123*	Yes	Yes	Limited	System development; still in draft form
	IEC Std Pub 880*	Yes	Yes	Yes	Intended for nuclear applications
	IEC 987*	Yes	Yes	Yes	Intended for nuclear applications
Institute of Electrical and Electronic Engineers (IEEE)	ANSI/IEEE 1012-1986*	Yes	Yes	Yes	Traditional V&V for S/W
	P1228 (Late Draft)*	Yes	Yes	Yes	S/W safety requirements
Department of Defense (DOD)	MIL-STD-882C*	Yes	Yes	Yes	System safety program plan
	DOD-STD-2167A*	Yes	Yes	Yes	S/W development only; currently being revised
U.S. Air Force	AF REG 122-9	Yes	Not well suited	Yes	Intended for nuclear weapon systems
	AF REG 122-10	Yes	Not well suited	Yes	Intended for nuclear weapon systems

* Selected for Detailed Assessment

TABLE 4-1. INITIAL SAFETY VERIFICATION/VALIDATION ASSESSMENT (cont.)

Organization	Initial Assessment Criteria				
	Standard(s) Involved In Safety V&V Process	Detail Available on Process/Activities	Applicable to RR & Other Fixed Guideway Applications	Applicable to Different Configurations/ Design Philosophies	Comments
Federal Aviation Administration (FAA)	AC 25.1309-1A	Yes	Not well suited	Yes	May be used (in the future) with SAE ARP 4754 & ARP 4761
	DO 178B*	Yes	Yes	Yes	
Society of Automotive Engineers (SAE)	SAE ARP 4754 (Early Draft)	Yes	Limited	Yes	System integration requirements for aircraft systems
	SAE ARP 4761 (Early Draft)	Yes	Limited	Yes	System safety assessment guidelines
National Aeronautics and Space Administration (NASA)	Software Safety Standard (Early Draft)	Yes	Yes	Yes	These represent several key safety related documents, but numerous other documents exist addressing NASA-wide and project-specific requirements and guidelines
	NHB 1700.1(V1-B) (NASA-wide requirements document)*	Yes	Yes	Yes	
	SSP 30309* (Space Station)	Yes	Yes	Yes	
Jet Propulsion Laboratory (JPL)	JPL D-576*	Yes	Limited	Yes	IV&V approach
	JPL D-10058	Yes	Yes	Yes	Not a NASA-wide accepted standard; research report only

* Selected for Detailed Assessment

TABLE 4-1. INITIAL SAFETY VERIFICATION/VALIDATION ASSESSMENT (cont.)

Organization	Initial Assessment Criteria				
	Standard(s) Involved In Safety V&V Process	Detail Available on Process/Activities	Applicable to RR & Other Fixed Guideway Applications	Applicable to Different Configurations/ Design Philosophies	Comments
Nuclear Regulatory Commission (NRC)	ANSI/IEEE 7-4.3.2-1982*	Yes	Yes	Yes	Current standard for NRC
	ANS/IEEE P-7.4.3.2 (Early Draft)	Yes	Yes	Yes	Will provide additional design/assessment requirements for nuclear applications
Medical Industry	Reviewer Guidance...For Devices Undergoing 510K Review (FDA)	Yes	Not well suited	Yes	S/W development and safety V&V guidelines and review (audit) process
	IEC 62 (Sec.) 69* (Late Draft)	Yes	Very Limited	Yes	Risk assessment/hazard analysis guidelines
AECL/CANDU Ontario Hydro	982 C-H69002-0001*	Yes	Yes	Yes	S/W engineering requirements; intended for nuclear applications
National Institute of Standards & Technology (National Bureau of Standards)	FIPS PUB 101	Yes	Yes	Yes	Traditional V&V and testing of S/W; early guideline document
	FIPS PUB 132	Yes	Yes	Yes	S/W V&V; endorses ANSI/IEEE 1012-1986
Electronic Industry Association (EIA)	SEB6-A*	Yes	Yes	Yes	Design requirements for munition applications

* Selected for Detailed Assessment

TABLE 4-1. INITIAL SAFETY VERIFICATION/VALIDATION ASSESSMENT (cont.)

Organization	Initial Assessment Criteria				
	Standard(s) Involved In Safety V&V Process	Detail Available on Process/Activities	Applicable to RR & Other Fixed Guideway Applications	Applicable to Different Configurations/ Design Philosophies	Comments
North Atlantic Treaty Organization (NATO)	STANAG 4404 (Early Draft)	Limited	Limited	Yes	Safety design requirements for munition systems; does not address V&V in any detail
	STANAG 4452 (Early Draft)	Yes	Limited	Yes	Safety assessment requirements for munition systems; development on hold
European Space Agency (ESA)	ESA PSS-01-40 Issue 2	No	Unknown	Unknown	Document not available
	ESA PSS-05-01 Issue 2[*]	Yes	Yes	Yes	Addresses S/W engineering aspects
Transport Canada	TP 10770E	Limited	Yes	Yes	Research report only on safety validation issues
Underwriters Laboratory	UL 1998[*] (Late Draft)	Yes--Limited	Yes	Yes	Contains only very general analysis requirements

[*] Selected for Detailed Assessment

The selection of organizations' safety V&V methodologies from Table 4-1 to be subjected to the detailed assessment was based in large part upon the existence and availability of an associated formal standard or guideline document. This is because, as discussed earlier, the selection of the "best" existing methodology (the main objective of the Base Task) is best realized if the methodology is well-documented and contains sufficient detail such that it can be compared with other methodologies. In the absence of formal standards or guideline documents, consideration was also given to any methodologies for which reasonably detailed information was obtained on the process/activities utilized. To be selected for further study, the methodologies also had to be generally applicable or well-suited to 1) railroad and other fixed guideway safety critical computer-based equipment, and 2) different computer system design philosophies.

Several of the standards identified during this study are drafts which are in various stages of development (i.e., awaiting comments from working groups and interested parties). In general, draft standards that were in the later stages of development at the time of this report were included in the detailed assessment only if they met the other criteria. On the other hand, early draft documents were not selected for the detailed assessment since development efforts were either just underway and/or significant changes were expected to be made to the draft. However, the information in the early draft standards will be useful during the Option Task.

By reviewing Table 4-1, it can be observed that except for the North American railway equipment suppliers and a few other firms for which no or limited information was received, standards or major guideline documents (in existing or later draft stages) were identified for most of the remaining organizations addressed. It should be noted that in several instances (e.g., ABB Signal, Matra Transport, RTRI) good information was received, but it was either confidential in nature or was not received in sufficient detail to permit proper comparison with others. Most standards were found to be generally applicable to the equipment of interest and generally applicable to different design philosophies.

In summary, all identified existing standards or those in later draft stages for which information was available and which could generally be applied to the equipment of interest and different design philosophies were selected for the detailed assessment. These are designated with an asterisk in Table 4-1.

4.2 DETAILED ASSESSMENT

The primary purpose of this detailed assessment was to further investigate the methodologies selected from the standpoints of their applicability and level of assured safety and to identify their attributes and limitations should they represent a railroad-specific safety validation standard. As described above, this assessment was directed to specific standards or guideline documents which define those methodologies and which met the other criteria of the initial assessment.

4.2.1 Detailed Assessment Criteria and Results

Criteria utilized in this assessment were directed to some general and applicability issues as well as the level of assured safety if a given methodology was to be applied in the safety evaluation of a computer-based system or item of equipment. The latter area included an investigation into the nature and comprehensiveness of each methodology in order to assess how well or thoroughly it addresses such safety concerns as hardware failures and software errors.

Results of the assessment relative to several general and applicability aspects of the methodologies are provided in Table 4-2. The nature of the information provided in Table 4-2 including the criteria utilized is summarized below:

- General Nature of Document(s) – (self-explanatory)
- Level of Application – the portion (i.e., overall system, hardware or software) of a computer-based system or item of equipment to which the standard applies
- When Applied – during what stage of the equipment's life cycle the verification/ validation process or activities are applied (e.g., during development, post development)
- Tailoring Required -- whether tailoring (e.g., selecting various activities that should be performed) is required when applying the methodology described in the document
- Coverage of Different Design Philosophies – how well the methodology covers different computer system design philosophies such as single and multiple channel systems with different types of hardware and/or software redundancy; usually designated in the table as Low, Medium or High (coverage)
- Coverage of Different Technologies – how well the methodology covers different types of software-based systems as well as other technologies such as hardware-only systems.

TABLE 4-2. RESULTS OF DETAILED ASSESSMENT — GENERAL AND APPLICABILITY ISSUES

Document	General Nature of Document(s)	Level of Application	When Applied	Tailoring Required	Coverage of Different Design Philosophies	Coverage of Different Technologies	Comments
ATCS Spec 140	Safety/system assurance program requirements	System, H/W & S/W	During development	Yes--extensive	High	High--not technology specific	For ATCS applications
ATCS Spec 130	Software development requirements	S/W	During development	No--already tailored from DOD-STD-2167	High	S/W only	S/W quality - assurance for ATCS applications
RIA Tech Spec No. 23	Software integrity requirements	S/W	During development	Yes--based in part on integrity levels	Medium--single channel systems not recommended for high levels of safety	S/W only	For railway applications
UIC 738R	General design and assessment guidelines	System, H/W & S/W	During or post development	Yes	High	S/W based systems only	For railway applications
UIC/ORE A155/RP11	Proof of safety (validation) recommendations	System, H/W & S/W	During or post development	Yes--extensive	High	S/W based systems only	For railway applications
UIC/ORE A155.1/RP8	Proof of safety recommendations	H/W & S/W	During or post development	Yes--extensive	High	S/W based systems only	Pertains to transmission systems only; for railway applications
DIN V VDE 0801	Detailed design and assessment guidelines	System, H/W & S/W	During development	Yes--extensive	High	S/W based systems only	Requires use of DIN V 19250
DIN V 19250	Qualitative risk assessment guidelines	System	Pre-development	No	High	High--not technology specific	To be used with DIN V VDE 0801

TABLE 4-2. RESULTS OF DETAILED ASSESSMENT — GENERAL AND APPLICABILITY ISSUES (cont.)

Document	General Nature of Document(s)	Level of Application	When Applied	Tailoring Required	Coverage of Different Design Philosophies	Coverage of Different Technologies	Comments
DIN VDE 0831	Installation requirements	System. H/W	Installation and operation of system	No	N/A	Available version not well suited to S/W based systems	Versions for S/W based systems may exist, but not available at time of study; for railway applications
Mü 8004	Development and proof-of-safety requirements	System. H/W & S/W	During development	No	High	High--not technology specific	DB standard
MOD 00-55 (Parts 1 and 2)	Software development requirements	S/W	During S/W development	Minimal	High	S/W only	To be used with MOD 00-56; military applications
MOD 00-56	Hazard analysis and risk assessment requirements	System and H/W	During entire system life cycle	Minimal	High	High, but specifically directed to S/W based systems	To be used with MOD 00-55; military applications
IRSE Report No. 1	Proof-of-safety recommendations	System. H/W. S/W	During development	Minimal	High	High	Also recommends use of IEC65A draft standards; for railway applications
IEC 65A (Sec) 123	System development and assessment requirements	System and H/W	During development	Yes--extensive; based on assigned integrity levels	Medium--single channel systems not recommended for high levels of safety	High	Not railway specific, but complementary to 65A (Sec) 122
IEC 65A (Sec) 122	Software integrity requirements	S/W	During development	Yes--extensive; based on assigned integrity level	Medium--single channel systems not recommended for high levels of safety	S/W based systems only	Not railway specific, but complementary to 65A (Sec) 123

TABLE 4-2. RESULTS OF DETAILED ASSESSMENT — GENERAL AND APPLICABILITY ISSUES (cont.)

Document	General Nature of Document(s)	Level of Application	When Applied	Tailoring Required	Coverage of Different Design Philosophies	Coverage of Different Technologies	Comments
IEC STD PUB 880	Software development requirements	S/W, some H/W	During development	Yes--extensive for S/W testing	High	S/W based systems only	To be used with IEC 987; for nuclear applications
IEC 987	H/W development requirements	H/W	During development	Yes	High	High	To be used with IEC 880; for nuclear applications
ANSI/IEEE 1012-1986	S/W V&V plan requirements	S/W	S/W life cycle	Yes--some	High	S/W based systems only	
IEEE P1228 (New Draft)	S/W safety plan requirements	S/W, some system	S/W life cycle	Yes--extensive, since only general requirements stated	High	S/W based systems only	Late draft
ANSI/IEEE - ANS-7-4.3.2-1982	Application criteria for computers (in nuclear power stations)	System, S/W and some H/W	During system development	Yes--only general requirements given	High	S/W based systems only	Currently being revised; for nuclear applications
MIL-STD-882C	System safety program plan requirements	System, H/W and S/W	System life cycle	Yes--extensive	High	High	Supersedes MIL-STD-882B with notice 1 and MIL-STD-1574A
DOD-STD-2167A	S/W development requirements	S/W	During development	Yes--extensive	High	S/W only	Currently being revised (MIL-STD-SDD)--in draft stage
DO 178B	S/W development guidelines	S/W	During S/W development	Yes--since in form of guidelines	Medium--use of dissimilar S/W is recommended	S/W only	Intended for aviation community
NHB 1700.1 (V1-B)	NASA-wide safety policy and requirements	System, H/W and S/W	During system development	Yes--extensive	High	High	Chapter 3 relates to system safety; for aerospace applications

TABLE 4-2. RESULTS OF DETAILED ASSESSMENT — GENERAL AND APPLICABILITY ISSUES (cont.)

Document	General Nature of Document(s)	Level of Application	When Applied	Tailoring Required	Coverage of Different Design Philosophies	Coverage of Different Technologies	Comments
SSP 30309	Safety analysis and risk assessment requirements	System, H/W and S/W	During system development	Yes--extensive	High	High	For NASA Space Station Freedom program
JPL D-576	Methodology for IV&V of computer S/W	S/W	During development	Yes--extensive	High	S/W only	Guidelines only; for aerospace applications
IEC 62 (Sec.) 69	Risk assessment guidelines	System	Early stages of development	Minimal	High	High	For medical applications
CLC/TC9X/ SC9XA/WGA2 (Draft 5)	Proof-of-safety requirements	System, H/W & some S/W	During development	Minimal	Medium--single channel systems not recommended	High	To be used with CLC/TC9X/ SC9XA/WGA1 (not currently available)
982 C-H 69 002-0001	S/W engineering requirements	S/W, some H/W	During development	Minimal	High	S/W based systems only	Applies to safety critical S/W; for nuclear applications
EIA SEB6-A	S/W safety guidelines	S/W primarily, but also system & H/W	During development	Yes--extensive	High	S/W based systems only	To be used with MIL-STD-882B and DOD-STD-2167A
ESA PSS-05-01 Issue 2	S/W engineering standards	S/W primarily, with some H/W	During development	Yes	High	S/W based systems only	For aerospace applications

TABLE 4-2. RESULTS OF DETAILED ASSESSMENT — GENERAL AND APPLICABILITY ISSUES (cont.)

Document	General Nature of Document(s)	Level of Application	When Applied	Tailoring Required	Coverage of Different Design Philosophies	Coverage of Different Technologies	Comments
UL 1998	Product investigation requirements	S/W primarily, with some H/W	During or post-development	Minimal	High	S/W based systems only	To be used with UL 991 for H/W

Results of assessing the methodologies relative to the level of assured safety are provided in Table 4-3. Brief descriptions of the criteria utilized in Table 4-3 are provided below:

- Nature of Techniques Involved – the types of safety verification/validation techniques required and/or recommended by the methodology; A = analysis, T = testing, S = simulation, M = modeling, DR = design review, R = review, C = calculation, I = inspection

- Involves Hazard/Risk Assessment – whether or not the process includes a hazard analysis and/or risk assessment, usually early in the development process

- Coverage of Normal Operation – how well the process covers or demonstrates safe operation of, especially, the hardware (and software) under normal operating conditions (e.g., in the absence of hardware failures); designated by L (low), M (medium), or H (high)

- Coverage of Hardware Failures – how well the process covers or demonstrates safe operation (of hardware portions of the system) in the event of hardware failures; designated by L, M, or H

- Coverage of H/W-S/W Interaction – how well the process covers or demonstrates safe execution of the software in the event of hardware failures; designated by L, M, or H

- Coverage of Latent Failures – how well the process covers or detects latent failures in, especially, the hardware; designated by L, M, or H

- Coverage of Common Mode Failures – how well the process covers or detects common mode failures in, especially, the hardware; designated by L, M, or H

- Coverage of Power Supply Anomalies/Transients – how well the process covers or demonstrates safe operation in the presence of power supply abnormalities and power transients; designated by L, M, or H

- Coverage of Improper/Abnormal Inputs – how well the process covers or demonstrates safe operation of hardware and software under conditions of abnormal or improper inputs (input signals outside the normal range); designated by L, M, or H

- Coverage of Human Interface – how well the process covers or demonstrates safe operation relative to human interaction with the system; designated by L, M, or H

TABLE 4-3. RESULTS OF DETAILED ASSESSMENT — LEVEL OF ASSURED SAFETY

Document	Nature of Techniques Involved	Involves Hazard/ Risk Assessment	Coverage of Normal Operation	Coverage of H/W Failures	Coverage of H/W-S/W Interaction	Coverage of Latent Failure	Coverage of Common Mode Failures	Coverage of Power Supply Anomalies/Transients	Coverage of Improper/ Abnormal Inputs	Coverage of Human Interface	Coverage of S/W Specification Errors	Coverage of S/W Design/Coding Errors	Coverage of Different Programming Languages	Involves Formal Methods for S/W	Involves Quantitative Assessment	Coverage of Data Transmission Aspects	Coverage of H/W & S/W Modifications	Coverage of Environmental Aspects	Comments
ATCS Spec 140	A,T, S,DR	Yes	H	H	M	L	L	M	H	H	M	M	M	No	No	M	L	M	Refers to MIL-STD-882B for Additional Guidance
ATCS Spec 130	T, DR	No	L	-	-	-	-	-	-	-	L	L	M	No	No	L	L	-	S/W quality assurance; ATCS Spec 140 provides safety assurance
RIA Tech Spec No. 23	A,T,S, M,DR	No	H	-	M	-	-	-	H	-	H	H	H	Rec	No	M	M (S/W)	L	Applies to S/W only; uses results of hazard/risk assessment from IEC 65A system standard
UIC 738 R	A,T, S,M	No	L	L	L	L	L	L	L	L	L	L	L	Rec	Opt	M	L	L	Describes very general design & assessment process/activities; must be used in conjunction with numerous other detailed ORE reports
UIC/ORE A155/RP11	A,T, S,M	No	H	H	H	H	M	M	L	L	L	H	H	Opt	Opt	L	L	H	To be used with other ORE design reports
UIC/ORE A155/RP8	A,T, S,M,C	No	H	H	H	H	M	L	L	L	L	L	M	No	Yes	H	L	L	To be used in conjunction with above report for S/W matters
DIN V VDE 0801	A,T, S	No	M	M	M	M	M	M	M	L	M	M	M	Opt	No	M	L	M	Uses results of risk assessment in DIN V 19250
DIN V 19250	A	Yes	L	L	L	L	L	L	L	L	L	L	-	-	No	L	-	L	Risk assessment only
DIN VDE 0831	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	No safety evaluation process described except for general testing requirements
Mü 8004	A,T	No	H	H	M	H	H	M	M	L	L	M	M	Opt	No	H	L	L	Numerous supplements exist to main document
MOD 00-55 (Parts 1& 2)	A,T, S,DR	No	H (S/W)	-	L	-	-	-	M	L	H	H	M	Yes	-	L	L	L	Addresses S/W only; uses results of hazard/risk assessment from MOD 00-56
MOD 00-56	A	Yes	L	L	L	L	L	L	L	H	L	L	-	-	No	L	L	L	Hazard/risk assessment only

TABLE 4-3. RESULTS OF DETAILED ASSESSMENT — LEVEL OF ASSURED SAFETY (cont.)

Documen.:	Nature of Techniques Involved	Involves Hazard/ Risk Assessment	Coverage of Normal Operation	Coverage of H/W Failures	Coverage of H/W-S/W Interaction	Coverage of Latent Failure	Coverage of Common Mode Failures	Coverage of Power Supply Anomalies/Transients	Coverage of Improper/ Abnormal Inputs	Coverage of Human Interface	Coverage of S/W Specification Errors	Coverage of S/W Design/Coding Errors	Coverage of Different Programming Languages	Involves Formal Methods for S/W	Involves Quantitative Assessment	Coverage of Data Transmission Aspects	Coverage of H/W & S/W Modifications	Coverage of Environmental Aspects	Comments
IRSE Report No. 1	A,T, S	Yes	H	H	H	L	H	L	L	L	H	H	H	Opt	Rec.	M	L	L	Must be used with IEC 65A
IEC65A (Sec.) 123	A,T, I,DR	Yes	H	H	L	M	H	L	H	K	H	-	-	-	Yes (H/W)	M	M (H/W)	M	Addresses system & H/W issues only
IEC65A (Sec.) 122	A,T, S,M	No	H	-	M	-	-	-	H	-	H	H	H	Opt	No	M	M (H/W)	M	Will be basis of new CENELEC S/W standard
IEC STD PUB 880	T,S, A	No	M	-	M	-	-	-	M	M	M	M	M	Opt	No	L	L	L	Addresses S/W & S/W - H/W interaction
IEC 987	A,T, DR	No	L	L	-	L	L	L	L	L	-	-	-	-	No	L	L (H/W)	L	Addresses H/W only
ANSI/IEEE 1012-1986	A,T	No	L (S/W)	-	L	-	-	-	L	L	M	M	H	No	No.	L	L (S/W)	L	Addresses S/W only
IEEE P1228 (New draft)	A,T, I	Yes	M (S/W)	-	L	-	-	-	L	H	M	M	H	No	Opt	L	L	L	Addresses some system aspects along with S/W
ANSI/IEEE-ANS-7-4.3.2-1982	T	No	L	-	L	-	-	-	L	L	L	L	H	No	No	L	L	L	Very early document
MIL-STD-882C	A,T,S, M,D	Yes	M	M	L	L	L	L	L	H	L	L	H	No	No	L	L	L	General requirements only
DOD-STD-2167A	A,T, DR	No	M (S/W)	-	L	-	-	-	L	L	M	M	H	No	No	L	L	L	Is to be used in conjunction with MIL-STD-882
DO 178B	A,T, R	No	H (S/W)	-	M	-	-	-	H	L	H	M	H	Opt	No	L	L	L	Addresses S/W with some attention to H/W - S/W interaction
NHB 1700.1 (V1-B)	A,T, DR	Yes	L	M	L	L	L	L	L	M	L	L	H	No	No	L	L	L	General requirements only
SSP 30309 (Rev B)	A,T	Yes	L	M	L	L	L	L	L	M	M	M	H	No	Opt	L	L	L	A newer version may exist

TABLE 4-3. RESULTS OF DETAILED ASSESSMENT — LEVEL OF ASSURED SAFETY (cont.)

Document	Nature of Techniques Involved	Involves Hazard/ Risk Assessment	Coverage of Normal Operation	Coverage of H/W Failures	Coverage of H/W-S/W Interaction	Coverage of Latent Failure	Coverage of Common Mode Failures	Coverage of Power Supply Anomalies/Transients	Coverage of Improper/ Abnormal Inputs	Coverage of Human Interface	Coverage of S/W Specification Errors	Coverage of S/W Design/Coding Errors	Coverage of Different Programming Languages	Involves Formal Methods for S/W	Involves Quantitative Assessment	Coverage of Data Transmission Aspects	Coverage of H/W & S/W Modifications	Coverage of Environmental Aspects	Comments
JPL D-576	A,T, R,S	No	M (S/W)	-	L	-	-	-	L	L	M	M	H	No	No	L	L	L	Independent V & V approach
IEC65 (Sec.) 69	A	Yes	L	L	L	L	L	L	L	L	L	L	-	-	No	L	L	L	Risk assessment process only
CLC/TC9X/SC9XA/WGA2 (Draft 5)	A,T	Yes	H	H	L	H	H	H	M	H	-	-	-	-	Yes (H/W)	M	M	H	S/W aspects including H/W - S/W interaction addressed in CLC/TC9X/SC9XA/WGA1 (draft)
982C-H 69002-0001	A,T	No	H (S/W)	-	M	-	-	-	M	L	H	M	H	No	No	L	L	L	Covers entire S/W engineering process; intended for nuclear applications
EIA SEB6-A	A,T	Yes	L (S/W)	-	L	-	-	-	L	M	M	M	H	No	Opt	L	L	L	Addresses S/W with some attention to H/W - S/W interaction
ESA PSS-05-01 Issue 2	A,T	No	L	-	L	-	-	-	L	L	M	L	H	Opt	No	L	L	L	Addresses S/W with some attention to H/W - S/W interaction
UL 1998	A,T	Yes	L	-	M	-	-	-	L	L	L	M	H	No	No	L	L	L	Addresses S/W with some attention to H/W - S/W interaction

- Coverage of Software Specification Errors – how well the process covers or detects specification/requirement type errors in the software; designated by L, M, or H

- Coverage of Software Design/Coding Errors – how well the process covers or detects errors as a result of detailed software design and coding; designated by L, M, or H

- Coverage of Different Programming Languages – how well the process can be applied to different programming languages including assembler and high level languages; designated by L, M, or H

- Involves Formal Methods for Software – whether the process requires or recommends the use of formal methods in the development of the software; usually designated by Yes, No, Rec (Recommended), or Opt (Optional)

- Involves Quantitative Assessment – whether the process requires or recommends a quantitative assessment of the level of safety achieved (e.g., mean-time-between-unsafe-failure); usually designated by Yes, No, Rec, or Opt

- Coverage of Data Transmission Aspects – how well the process covers or demonstrates the safe transmission of data under different operating conditions (i.e., normal and failure/abnormal conditions); designated by L, M, or H

- Coverage of Hardware and Software Modifications – how well the process covers or demonstrates safe operation following hardware and/or software modifications; designated by L, M, or H

- Coverage of Environmental Conditions – how well the process covers or demonstrates safe operation under anticipated environmental conditions; designated by L, M, or H.

It is necessary to discuss two major issues regarding the assessment results in Table 4-3. First, the focus in assessing the level of safety was on the effectiveness of a given methodology in helping to ensure the safety of a computer-based system (including hardware, software and other aspects) as opposed to just the safety of the software. Second, the criteria utilized in Table 4-3 to assess the level of safety is not intended to be a comprehensive or exhaustive list of all necessary characteristics of a good methodology. Rather, the criteria were selected to help give a general indication as to the completeness and overall effectiveness of the methodology and to establish a baseline for comparison.

4.2.2 Detailed Assessment Summary

Based upon the assessment results in Tables 4-2 and 4-3, it was possible to compare the various methodologies both as to their applicability and general level of assured safety. Summaries of the methodologies from these two standpoints are provided below. In addition, various attributes and limitations of each methodology are identified.

4.2.2.1 Applicability - Overall, the applicability of the methodologies assessed was found to be generally high. That is, most could be applied to the general types of safety critical computer-based systems/equipment (e.g., speed control, train detection, interlocking) which may be experienced in U.S. applications.

Also, most could be applied to equipment with different design philosophies (e.g., single or multiple channel configurations with different levels of software redundancy or types of software safety features). Exceptions include those methodologies developed by the RIA (i.e., RIA Tech Spec No. 23), IEC (i.e., IEC 65A Sec 122 and 123), RTCA (i.e., DO 178B) and CENELEC (i.e., CLC/TC9X/SC9XA/WGA2). These organizations generally do not recommend the use of single channel computer-based systems for "highly" safety critical applications, where highly in this context refers to systems with the highest possible level of safety integrity or with the highest associated risk (e.g., interlocking system).

A great deal of diversity was observed relative to the extent of system coverage of the methodologies. For example, some (approximately one-half) were applicable to an entire system (including hardware, software and interfaces such as hardware-software and human interfaces). Others (approximately one-third) applied only to software. Only one of the methodologies addressed (computer) hardware only (i.e., IEC 987), but it was intended to be used in conjunction with another document (i.e., IEC 880) directed primarily to software.

In terms of when the methodologies are applied, most were found to be structured such that they are utilized at one or more points in the system or software development process rather than just at the end of development.

Many of the methodologies (i.e., about one-half) were found to require extensive tailoring. In some cases this was because the methodology presented a "menu approach," where a supplier could select from a variety of different activities or analyses/tests to be performed. This was also associated with methodologies that included a means of classifying the system or functions thereof into different levels of safety (i.e., safety integrity levels) depending upon the degree of risk in the system. Certain activities and techniques were recommended for different integrity levels. It should be noted that one reason for the use of integrity levels in some of the methodologies is the increasing complexity of computer-based systems and the cost of developing and proving the safety of large complex systems. The use of different integrity levels for different systems or functions allows one to focus development/assessment efforts on specific systems or smaller portions of those systems.

In other cases, tailoring was due to the nature of the methodology being in the form of very general guidelines or requirements where a supplier has the option of establishing their own activities and/or analysis/testing techniques.

4.2.2.2 Level of Assured Safety - No one single methodology "stood out" as being highly comprehensive in terms of assuring/demonstrating the safety of a complete system (i.e., hardware, software and interfaces). In some cases where the coverage of certain aspects was judged to be "high" or good, the methodology did not cover an entire system (e.g., covered only software or certain system aspects). In other cases where an entire system and its interfaces were addressed, the methodology did not fully cover or address certain aspects (e.g., hardware failure effects upon software execution).

Overall, a great deal of diversity was observed in the assessment of the methodologies relative to their level of assured safety. The following comments summarize this aspect (i.e., level of assured safety) of the assessment and reflect the associated diversity of the methodologies from this standpoint:

1) A variety of verification/validation activities (e.g., analysis, testing, simulation, modeling) are being required/recommended, but most are based upon (at least) a combination of analysis and testing. A great deal of variation also exists in the specific techniques utilized within those activities. For example, a wide number of different hardware, software and system analysis techniques are being used such as FMEAs, Fault Trees, Sneak Circuit Analysis, Petri Nets, Common Cause Failure Analysis, Markov Modeling, Formal Methods and many others.

2) Just less than half of the methodologies require/recommend the conduct of hazard analyses and risk assessments during the early design stages to identify and eliminate/minimize potential risks as early in the design as possible.

3) The coverage or demonstration of safe operation under conditions of normal operation varies greatly. Some of the methodologies directly specify this coverage while others do not address it at all. As reflected here, it is an often overlooked aspect of safety assurance, especially from a hardware standpoint.

4) Coverage of hardware failures, for those methodologies addressing hardware aspects, is generally high.

5) Coverage of the hardware/software interaction (e.g., hardware failure effects upon software execution) is generally low to medium.

6) Coverage of latent failures, common mode failures, and power supply anomalies/transients is generally low.

7) Coverage of improper/abnormal inputs and human interface aspects varies greatly. Some methodologies cover the areas well while others do not address them at all.

8) Coverage of software specification, design and coding errors varies greatly. To no surprise, those methodologies directed specifically and exclusively to software do the best job of addressing these aspects.

9) Most methodologies can be utilized with different (including higher level) programming languages.

10) Most methodologies do not require the use of formal methods (in the development of software). The one exception is the standard MOD 00-55, developed by the U.K. Ministry of Defence. About one-fourth of the methodologies do acknowledge the existence of formal methods, but make the application optional.

11) Most methodologies do not require the use of quantitative safety assessments to supplement other safety verifications/validations. The exceptions are the UIC/ORE (for data transmission systems), IEC (for hardware) and CENELEC (for hardware).

12) Coverage of data transmission aspects is generally low, except for the UIC/ORE which has developed a document to specifically address these aspects. It should be noted, however, that even though data transmission is not specifically cited in many of the methodologies, it can be addressed (in part) by many of the same verification/validation practices and techniques utilized on other portions of the computer system.

13) Coverage of hardware and software modifications and associated reverifications/revalidations is generally very low for all methodologies, including those associated with the rail industry. While many methodologies provide general statements that the impact of modifications should be checked or confirmed, they do not provide specifics on how this should be done. In many cases, it is recommended that an entire retest and/or reanalysis is performed on the entire piece of equipment.

14) Coverage of environmental aspects is generally very low. Many methodologies do not address the issue at all. Two exceptions are the UIC/ORE and CENELEC.

It should be noted that it was somewhat difficult to assess the effectiveness and compare more general methodologies with those that provide more detail or present "menus" of activities/techniques. However, every attempt was made to assess the "best case" situation should the methodology be applied.

4.2.2.3 Attributes and Limitations - By reviewing the assessment results relative to applicability and level of assured safety of the methodologies and the documents themselves (which describe the methodologies), it was possible to identify various attributes and limitations of each methodology. These are summarized in Table 4-4 together with key features of each methodology.

TABLE 4-4. ATTRIBUTES AND LIMITATIONS OF RELEVANT SAFETY V&V RELATED STANDARDS

Document	Key Features	Attributes	Limitations	Other Comments
ATCS Spec 140	• Specifies System Safety Plan requirements • Defines minimum essential safety analysis (hazard analyses) tools and techniques during development • Requires verification testing to verify results of safety analyses	• Addresses wide range of system hazards • Good general coverage of H/W and S/W aspects • Good coverage of human interface concerns and effects of abnormal inputs	• Directed to ATCS development/ implementation • Requires extensive tailoring for specific applications • May be modified based on new S/W procurement concept	• Refers to MIL-STD-882B for further guidance • Requires separate quality assurance plan • Independent V&V recommended
ATCS Spec 130	• Defines software QA process and requirements	• Defines relatively low cost S/W QA process • Supports detection and correction of S/W errors	• Not specifically directed to S/W safety--but helps to achieve S/W safety. • Must still use ATCS Spec 140 to help ensure safety	• Tailored from DOD-STD-2167

TABLE 4-4. ATTRIBUTES AND LIMITATIONS OF RELEVANT SAFETY V&V RELATED STANDARDS (cont.)

Document	Key Features	Attributes	Limitations	Other Comments
RIA Tech Spec No. 23	• Defines S/W design and assessment requirements for railway applications • Provides guidance on defining integrity levels for railway applications • Requires verification and test after each stage of development • Requires independent validator and assessor	• Pertains exclusively to railway applications • Relatively comprehensive for S/W aspects--identifying S/W errors • Uses wide range of techniques for assessing safety	• Applies to S/W only • Limited applicability to single channel systems • Difficult to apply V&V unless associated development process is followed	• Provides interpretation of earlier draft (94) of IEC65A (Sec) 122 • Requires quality assurance plan • Recommends design techniques and documentation • To be expanded to include system and H/W aspects • British Rail and railway equipment manufacturers on technical committee
UIC 738R	• Describes general design and assessment guidelines for computer systems in railway applications • Recommends independent safety audit	• Relatively comprehensive validation process described • Can be applied during or after development	• Document is too general to use by itself	• Numerous other UIC/ORE reports are referenced

TABLE 4-4. ATTRIBUTES AND LIMITATIONS OF RELEVANT SAFETY V&V RELATED STANDARDS (cont.)

Document	Key Features	Attributes	Limitations	Other Comments
UIC/ORE A155/RP11	• Describes proof of safety process. based on validation (for railway computer systems) • Recommends independent safety audit • Describes separate H/W. S/W and system validations	• Specifically directed to safety validation • Covers entire system • Relatively comprehensive validation process • Can be applied during or post development	• Guidelines only--requires extensive tailoring • Weak on addressing effects of improper inputs to system and human interface concerns • Weak on coverage of specification errors--may be addressed in other ORE reports	• Numerous other ORE reports address design issues
UIC/ORE A155.1/RP8	• Describes proof of safety for S/W based transmission systems	• Good discussion and coverage of transmission system issues	• Extensive tailoring required	• Must be used in conjunction with A155/ RP11 and other reports to ensure safety
DIN V VDE 0801	• Discusses possible design and assessment measures to avoid. detect and control failures/errors • Provides some guidance on selection of measures based in part on assigned requirements classes	• Extensive discussion on nature and benefits of design/assessment measures • Applies to system. H/W and S/W • Good discussion/treatment of H/W and S/W concerns in computer systems	• Extensive tailoring required • Emphasis on design (i.e.. error avoidance) • Difficult to apply assessment process/ measures unless associated design process is followed	• Will be superseded by future European standards (e.g.. CENELEC) • Must be used in conjunction with DIN V 19250 for assignment of requirements classes • Quality assurance process also suggested
DIN V 19250	• Describes qualitative risk assessment procedure • Defines 8 requirements classes/categories	• Pertains to railway and other applications	• Only very general risk assessment process described	• Helps determine requirements classes for use with DIN V VDE 0801

TABLE 4-4. ATTRIBUTES AND LIMITATIONS OF RELEVANT SAFETY V&V RELATED STANDARDS (cont.)

Document	Key Features	Attributes	Limitations	Other Comments
DIN VDE 0831	• Describes railway signalling installation requirements	• Specific to railway industry	• Contains no safety assessment process other than some general testing requirements • Current available version not well-suited to S/W based systems	• New S/W based version may exist, but probably still related to installation requirements
Mü 8004	• Provides safety requirements for development • Defines structure/content of proof-of-safety document • Requires separate and independent validation	• Defines specific safety analysis requirements • Extensive treatment of H/W failures including failure mode list • Good treatment of data transmission aspects • Good treatment of independence/common mode failure issues	• Only general requirements for S/W assessment (testing)--specific techniques not identified • Only general requirements for showing safety of H/W-S/W interaction--specific concerns/analyses/tests not identified • English translation difficult to interpret in places	• Numerous supplements are being added--many address specific S/W application-specific issues • Requires quality assurance process (e.g., EN29000/ISO 9000)
MOD 00-55 (Parts 1 & 2)	• Describes requirements for procurement/development of safety critical S/W • Requires verification at each stage of development • Requires overall validation testing • Uses formal (method) development approach	• Specifically intended for development of safety critical S/W • Good coverage of S/W spec. design and coding errors	• Addresses S/W only • Minimal coverage of S/W-H/W interaction	• To be used with MOD 00-56 • Requires quality assurance plan (e.g., ISO 9000) • Requires independent safety auditor

TABLE 4-4. ATTRIBUTES AND LIMITATIONS OF RELEVANT SAFETY V&V RELATED STANDARDS (cont.)

Document	Key Features	Attributes	Limitations	Other Comments
MOD 00-56	• Describes systems approach to identify hazards, determine risks and assign safety integrity levels	• Good systems approach to identify wide variety of hazards • Minimal tailoring required	• Does not address verification or validation of H/W, S/W or system	• To be used with MOD 00-55 • Requires quality assurance plan (e.g., ISO 9000) • Documentation requirements specified • Requires independent safety auditor
IRSE Report No. 1	• Provides recommendations for proof-of-safety of railway signalling systems--includes use of IEC 65A draft standards • Suggests verification after each phase • Suggests overall system validation • Suggests independent safety assessment	• Applies to signalling systems • Stresses "total railway" safety • Cites separate and mandatory procedures for H/W, S/W and system assurance (they are relatively comprehensive) • Good basis for a new standard	• A report in the form of recommendations only--cannot easily serve as a standard in current form • Methods/procedures cited are not clearly tied to verification/validation activities or with IEC 65A	• Indicates that no complete existing standard exists--but IEC 65A is one of the best • Requires quality assurance plan (e.g., EN29001)

TABLE 4-4. ATTRIBUTES AND LIMITATIONS OF RELEVANT SAFETY V&V RELATED STANDARDS (cont.)

Document	Key Features	Attributes	Limitations	Other Comments
IEC 65A (Sec) 123	• Provides requirements for system development • Safety evaluations discussed in context of safety life cycle • Includes hazard analysis/risk assessment • Verification required after each life cycle phase • Overall validation required • Requires independent functional safety assessment • Requires safety plan	• Relatively comprehensive for system and H/W safety aspects • Good general process for evaluation of H/W modifications	• Not railway specific, but provides good basis for application-specific standard • Still in draft form • Limited applicability to single channel systems • Difficult to apply validation process unless associated development process is followed • Requires extensive tailoring	• Must be used with IEC 65A (Sec) 122 • Requires quality plan (e.g., ISO 9000)
IEC 65A (Sec) 122	• Provides S/W integrity requirements for different levels of safety integrity • Requires (independent) S/W verification after each life cycle phase • Requires overall independent S/W validation	• Relatively comprehensive for S/W safety aspects--S/W errors • Good general process for evaluation of H/W modifications	• Not railway specific • Still in draft form • Limited applicability to single channel systems • Does not specify criteria for allocation of safety integrity levels for railway applications	• Uses hazard/risk assessment results from IEC 65A (Sec) 123 • Requires quality assurance plan (e.g., ISO 9000) • Requires use of coding standards

TABLE 4-4. ATTRIBUTES AND LIMITATIONS OF RELEVANT SAFETY V&V RELATED STANDARDS (cont.)

Document	Key Features	Attributes	Limitations	Other Comments
IEC STD PUB 880	• Describes S/W development and assessment requirements (for nuclear applications) • Requires S/W verifications after each life cycle phase • Requires overall computer system validation	• Reasonably good S/W development process for safety applications • Good treatment of H/W-S/W interactions	• Evaluations based almost entirely on S/W testing • Little detail on analysis techniques • Extensive tailoring required	• Used with IEC 987 • Requires parallel quality assurance plan
IEC 987	• Describes H/W development and assessment requirements • Requires "analysis of failures" and system safety analysis • Requires H/W verification after each design phase	• Addresses wide range of issues for computer-based H/W development	• Limited information on required safety analyses for hardware and systems • Directed to H/W only	• Used with IEC 880 for S/W aspects • Requires H/W quality assurance plan
ANSI/IEEE 1012-1986	• Provides requirements for content/format of S/W V&V plan	• Describes minimum V&V tasks for "critical" S/W • Provides relatively comprehensive set of requirements for S/W V&V	• V&V activities not specifically directed to safety • Provides only general plan requirements--details on techniques not included • H/W-S/W interaction not addressed	• A new S/W safety plan standard is being prepared • Other related standards address quality assurance, configuration management, documentation, etc.

TABLE 4-4. ATTRIBUTES AND LIMITATIONS OF RELEVANT SAFETY V&V RELATED STANDARDS (cont.)

Document	Key Features	Attributes	Limitations	Other Comments
IEEE P1228 (New Draft)	• Provides requirements for content of S/W safety plan • Requires S/W safety analyses	• Ties in S/W safety activities with system aspects via hazard analyses • Includes several S/W safety analyses to identify errors that could lead to hazards	• Provides general requirements only--specific analysis techniques not recommended • H/W-S/W interaction not addressed	• Draft is in later stages • Describes how other S/W aspects (e.g., quality assurance, V&V) relate to the safety program
ANSI/IEEE-ANS-7-4.3.2-1982	• Provides general requirements for S/W development and computer system validation • Requires verification at each stage in development	• Requires H/W-S/W validation as a system	• Very general requirements only--no evaluation techniques suggested or recommended • All validation based on testing only	• H/W development covered in IEEE STD 603-1980 • Currently being revised
MIL-STD-882C	• Provides requirements for developing and implementing system safety program plan • Requires numerous hazard analyses (e.g., PHA, SSHA, SHA) • Requires separate safety assessment • Requires separate safety verification (test/ demonstration)	• Excellent general safety requirements for a system --good coverage of system safety aspects • Multiple safety evaluations required throughout development • Ties in S/W development activities to system safety effort • Extensive guidance on selection and timing of tasks	• Only general requirements stated--specific analysis/ assessment techniques not identified • Extensive tailoring required	• Lack of detail on techniques makes it hard to assess coverage of H/W failures and S/W errors

TABLE 4-4. ATTRIBUTES AND LIMITATIONS OF RELEVANT SAFETY V&V RELATED STANDARDS (cont.)

Document	Key Features	Attributes	Limitations	Other Comments
DOD-STD-2167A	• Defines requirements for acquisition, development and support of S/W • Provides menu of deliverables, analyses/tests, reviews and audits that can be used	• Relatively comprehensive S/W development requirements • Would improve overall quality of S/W and its documentation	• Does not identify safety as major concern • Cites only very general requirement for safety analysis--no details provided • Requires extensive tailoring	• New draft under development places more emphasis on safety
DO 178 B	• Describes S/W development guidelines (for airborne systems and equipment) • Describes S/W verification process--to assess vaults (identify errors) of S/W development process via analysis, reviews and testing • Defines different S/W levels based on level of safety required	• Relatively comprehensive S/W verification (assessment) process with emphasis on safety • Good coverage of S/W specification type errors • Good discussion on types of S/W issues and errors to consider	• Describes activities and objectives, but not specific assessment techniques • No analyses for H/W-S/W integration--based entirely on testing	• Will relate to system safety assessment process guidelines--currently under development by SAE • Suggests use of coding standards • Suggests establishment of S/W quality assurance plan
NHB 1700.1 (V1-B)	• Chapter 3 addresses system safety • Requires safety management plan • Requires system safety analyses	• Good general system level safety requirements for ensuring safety • Multiple hazard analyses conducted throughout development	• Only general analysis requirements provided (with a menu of a few analysis techniques) • Details not provided on addressing H/W failures or S/W errors (this is nature of document)	• Forms basis for more detailed NASA safety-related requirements document • Similar in nature (but in less detail) to MIL-STD-882B

TABLE 4-4. ATTRIBUTES AND LIMITATIONS OF RELEVANT SAFETY V&V RELATED STANDARDS (cont.)

Document	Key Features	Attributes	Limitations	Other Comments
SSP30309 (Rev B)	• Provides safety analysis and risk assessment requirements for Space Station Freedom program • Requires safety analysis to identify and classify hazards • Requires risk assessment to verify safety goals are met	• Good general system level requirements for ensuring safety • Relatively comprehensive S/W analysis requirements	• Extensive tailoring required • Effects of H/W-S/W interaction not addressed • Details not provided on addressing various other H/W related aspects	• Based on NASA-wide requirements in NHB 1700.1 • Newer version (i.e., Rev C) may exist
JPL D-576	• Describes general approach to independent V&V of computer S/W • Describes inputs, process, outputs and tools for various IV&V tasks	• Relatively comprehensive IV&V approach	• Safety is not identified as a primary concern • Relatively old document • Addresses IV&V only	• Written primarily for aeronautical/space S/W applications
IEC 65 (Sec) 69	• Provides guidelines for conducting hazard analysis/ risk assessment (of medical equipment)	• Good general approach for medical applications	• Addresses general hazard analysis/risk assessment process only--no details on safety evaluation/assurance techniques • Not particularly well-suited to RR applications	• Refers to other documents for techniques (e.g., MIL-STD-882B, IEC 65A (Sec) 122 and 123, DIN V VDE 0801)

TABLE 4-4. ATTRIBUTES AND LIMITATIONS OF RELEVANT SAFETY V&V RELATED STANDARDS (cont.)

Document	Key Features	Attributes	Limitations	Other Comments
CLC/TC9X/ SC9XA/WGA2 (Draft 5)	• Defines requirements, conditions and evidence for safety acceptance of railway signalling systems--proof of safety (generic) or safety case (specific application) • includes hazard/risk assessment • Requires separate and independent safety assessment	• Directly addresses required evidence for demonstrating safety of railway systems/equipment • Very comprehensive set of evidence required--quality management, safety management and technical safety • Excellent coverage of system and H/W safety issues (e.g., safe normal operation, failures) • Excellent coverage of environmental issues	• Still in draft form and incomplete in some sections • Not applicable to all configurations since single channel systems not recommended--difficulty in applying directly in U.S. • Some design requirements specified may not be applicable in U.S. • Document specifies general requirements only--specific techniques generally not identified	• Additional S/W requirements identified in CLC/TC9X/SC9XA/ WGA1--must also be used • Quality plan required according to ISO 9001 or EN 29001 • Requires use of IEC 65A (Sec) 123 for defining safety integrity levels • Requires use of CENELEC TC9X-WG5B document for safety management and other purposes
982 C-H 69002-0001	• Provides S/W engineering requirements (for nuclear applications) • Define set of S/W engineering processes, set of associated outputs, and requirements for content of outputs • One process is verification--applies to requirements, design & code and includes validation testing	• Applies to safety critical S/W • Covers overall S/W quality aspects of which safety is just one • Relatively comprehensive S/W engineering process requirements	• Specific analysis/test techniques not identified • Trial document only	• Intended for nuclear applications, but applicable to other industries as well

TABLE 4-4. ATTRIBUTES AND LIMITATIONS OF RELEVANT SAFETY V&V RELATED STANDARDS (cont.)

Document	Key Features	Attributes	Limitations	Other Comments
EIA SEB6-A	• Provides guidelines on safety activities during S/W development via DOD-STD-2167A • Implements intent of MIL-STD-882B for S/W	• Good discussion of software safety issues • Relatively comprehensive hazard analysis & tracking process for S/W development	• Extensive tailoring required • Somewhat outdated with release of MIL-STD-882C • General treatment of requirements and techniques to validate S/W	• Intended for weapon system applications
ESA PSS-05-01 Issue 2	• Describes mandatory and recommended practices and guidelines for S/W engineering projects • Includes section on S/W V&V	• Good general discussion of S/W V&V	• Not specifically directed to safety • Describes only general V&V requirements	• Based heavily upon IEEE 1012 and other standards • Requires separate S/W QA program
UL 1998	• Provides product investigation requirements for S/W • Includes underlying S/W design requirements • Requires risk analysis, FMEA, FTA, and S/W code level analyses	• General but relatively comprehensive code level analysis requirements • Good emphasis on H/W-S/W interaction	• Describes only very general investigation/ analysis requirements • Minimal to no coverage of abnormal input conditions	• Intended for process control industry • Is to be used with end-product requirements for system investigation and UL 991 for H/W investigation

5. OVERALL SUMMARY

This portion of the report, separated into four main sections, provides an overall summary of, primarily, the methodology assessment. Following some general observations and a discussion on the diversity of the methodologies, there is a discussion on the recommendations as a result of the assessment. The section ends with the identification of certain trends that have been observed.

5.1 GENERAL OBSERVATIONS

A total of almost 60 major standards or guideline documents, which contained safety-related verification/validation methodologies utilized and/or developed by a wide variety of industries worldwide, were subjected to an initial assessment. Approximately one-half of these were then subjected to a more detailed assessment from the standpoint of applicability and level of assured safety.

It was found that the North American railway suppliers have and utilize (almost exclusively) their own internal standards and processes relative to safety verifications and validations. On the other hand, most European railway suppliers and authorities typically use one or more national standards plus their own internal standards/guidelines, many of which have been created to implement the intent of the national standards. There are certainly exceptions. In Sweden, for example, there are no national standards in this area. In Germany, one of the primary standards for the German Federal Railway (DB) is the document Mü 8004, which was developed by the DB. Although British Rail tends to generally follow the RIA Tech Spec No. 23, they have their own internal standards for verification and validation.

Interest is certainly great worldwide by all industries in this topic area as reflected by the numerous documents that exist or are in various stages of development. Some examples of draft standards that address safety verifications/validations are as follows:

- CENELEC CLC/TC9X/SC9XA/WGA1 - "Railway Applications: Software for Railway Control and Protection Systems"
- CENELEC CLC/TC9X/SC9XA/WGA2 - "Railway Applications: Safety Related Electronic Control and Protection Systems"
- IEC 65A (Sec) 122 - "Software for Computers in the Application of Industrial Safety Related Systems"
- IEC 65A (Sec) 123 - "Generic Aspects: Functional Safety of Electrical/Electronic/Programmable Safety Related Systems, Part 1, General Requirements"
- IEEE P1228 - "Standard for Software Safety Plans"

- ANSI/ANS 7-4.3.2 - "Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations"

- SAE ARP 4761 - "Safety Assessment Guidelines for Civil Airborne Systems and Equipment"

- NASA - "Software Safety Standard"

- IEC 62 (Sec) 69 - "Electrical Equipment in Medical Practice"

- UL 1998 - "Standard for Safety Related Software"

- NATO STANAG 4452 - "Safety Assessment of Munition Related Computing Systems."

It should also be noted that Europe and the international community is well ahead of the U.S. in creating standards/guidelines for safety critical computer-based systems, particularly in the railway industry. This is apparent in several ways, perhaps most obviously in the UIC/ORE design and assessment recommendations for computer-based systems. The UIC and ORE (now ERRI) has been working in this area since the 70's. Another example is the current work being conducted by CENELEC for the railway industry within the European Community. Two standards are being developed (one directed to software and the other to system/hardware aspects). This work has been underway for several years by a working group of numerous individuals from all over Europe.

Another observation is that terminology in the area of safety verifications and validations varies greatly among industries, organizations and individuals. In some instances the term safety verification was used to denote all activities that are performed to demonstrate the safety of a system. In other instances a safety verification was used to denote the activities performed at the end of each development phase to demonstrate compliance with the requirements of that phase – this is more consistent with the term verification in this country. A similar variation was found in the usage of the term safety validation. In at least one instance, safety validation was referred to as a type of audit in which someone or a group of individuals independently reviews the safety activities/assessments performed by others. Quality assurance and software quality assurance were also used in a variety of ways.

Perhaps the most significant observation concerns the diversity found in the various methodologies. This is addressed in more detail below.

5.2 DIVERSITY

A great deal of diversity exists in the methodologies included in this study. To begin with, the methodologies were developed (often within working groups) by personnel from diverse backgrounds and organizations (e.g., regulatory agencies, research firms, equipment suppliers, users) for different industries (e.g., rail, avionics, aerospace, military, nuclear, medical and consumer products). This in itself tends to diversify the methodologies since the objectives, the ways in which safety is viewed and the means of achieving safety differ from individual to individual, organization to organization and industry to industry.

The system coverage (or the portions or aspects of a system to which a given methodology applies) varies greatly. For example, some methodologies apply only to high level system aspects (e.g., conduct of system risk assessment) and do not directly address assessment of hardware or software. Two such methodologies are MOD 00-56 and DIN V 19250. Other methodologies (i.e., DO 178B, ANSI/IEEE 1012) apply only to software (development and assessment), while others address software and hardware or a combination of software, hardware and system aspects in varying degrees.

Further, the nature of the methodologies themselves varies tremendously. They range from requirements (either high or low level), guidelines and recommendations to menus of activities and techniques. Further, most methodologies specify "what is to be done" as opposed to "how to do it." Below are listed examples of the diversity that exists in the methodologies assessed:

- General system design and assessment guidelines (UIC 738R)
- Detailed design and assessment guidelines that describe means for detecting, avoiding and controlling errors and failures (DIN V VDE 0801)
- Software development requirements that include verification and validation aspects (DO 178B)
- Software verification and validation plan requirements (IEEE 1012)
- Software safety plan requirements (IEEE P1228)
- System safety program plan requirements for developing and implementing a system safety program (MIL-STD-882C, ATCS Tech Spec 140)
- System installation requirements (DIN 0831) – little safety verification and validation content
- System risk assessment requirements/guidelines for determining safety integrity levels (DIN V 19250, MOD 00-56)
- System/hardware proof-of-safety requirements addressing technical and management issues (CENELEC)
- Independent verification and validation (IV&V) guidelines (JPL D-576, AFSC 800-5)
- Transmission system proof-of-safety recommendations (ORE A155.1/RP8)

- Product investigation requirements that include system, hardware and software analyses (UL 1998)

- Software development requirements that include formal methods with assessments (MOD 00-55)

- Software engineering requirements that include verification and validation aspects (982 C-H69002-0001).

There is also diversity in the activities and the specific assessment techniques required or recommended by the different methodologies.

5.3 RECOMMENDED METHODOLOGY

The initial objective in this Base Task was to identify and describe existing safety verification/validation methodologies, assess them from the standpoint of their applicability and level of assured safety, and subject one or more of them to a techno-economic feasibility study after which a selection of the "best" existing methodology would be made. Further efforts in the Option Task were to be directed to developing a specific and industry-approved methodology based upon the results of the Base Task.

Although the majority of the methodologies assessed in this program were found to be generally applicable to equipment of interest and different design philosophies, each was found to "fall short" in some aspect(s) relative to assuring safety of a computer-based system. For example, some methodologies apply to software only as opposed to an entire computer system. Others were found to not fully cover or address certain types of hardware and/or software concerns. For these reasons, a single "best" existing methodology is not being recommended at this time.

Even though an existing methodology is not being recommended, it was considered beneficial to identify some of the methodologies which have significant attributes or qualities from, especially, a system safety verification/validation standpoint. Those are as follows (in no particular order of importance):

- ATCS Spec 140

- UIC/ORE A155/RP11 and A155.A/RP8

- Mü 8004

- MOD 00-55 and 00-56

- IEC 65 A (Sec.) 122 and 123

- MIL-STD-882C

- CENELEC CLC/TC9X/SC9XA/WGA1 and 2

Two others with particularly good attributes from just a software safety verification/ validation standpoint are RIA Tech Spec No. 23 and DO 178B. Several other software-related standards (e.g., IEEE 1012) were found to be quite extensive from a verification and validation standpoint, but not exceptionally strong in or particularly directed to safety issues.

The various attributes identified for these and other methodologies/standards are addressed in the Detailed Assessment Section of this report in Table 4-4.

It is anticipated that the attributes of the various methodologies described and assessed in this portion of the program will be utilized in addressing some of the shortfalls observed in other methodologies and in developing and recommending a reasonable and effective methodology for FRA's consideration.

5.4 TRENDS

As a result of reviewing the various existing methodologies and those in different stages of development across a number of industries worldwide (e.g. railroad, avionics, nuclear, military, medical, consumer product), a number of trends can be observed. Several of those are described below:

1) Safety-related assessments are being required/recommended throughout the development cycle of a computer-based system, from conceptual design through final development stages. Most include safety-related verifications following each major design phase of the system, and software and safety validations at the end of development.

2) Hazard analyses and risk assessments are being required/recommended in early design stages to help identify and eliminate (or reduce the risk associated with) potential system hazards and assign safety integrity levels to entire systems and/or specific functions.

3) A wide mix of analysis and testing techniques are being required/recommended – no clear choices are dominating.

4) (A "non-trend") - There is actually no clear trend toward either requiring or just recommending/suggesting possible verification/validation techniques. Some methodologies require specific techniques while others provide menus of techniques.

5) Emphasis has been on software, but is now becoming more comprehensive from a system standpoint as groups and organizations realize the importance of safety in a system context.

6) Formal methods for software development are gaining acceptance and are being recognized as useful techniques. To date, most methodologies do not require their use.

7) Methodologies are requiring/recommending separate safety-related development and assessment processes/activities for software (in addition to those for an overall system).

8) Methodologies are requiring/recommending independent safety assessments (to independently assess safety of equipment) and/or safety audits (to review safety activities and associated outputs conducted by others).

9) Methodologies are requiring/recommending the establishment and implementation of quality assurance plans (e.g., those associated with ISO 9000 or EN 29000 series standards) in addition to safety plans. The proper implementation of quality assurance plans is expected to minimize the existence of both hardware failures and software errors.

10) Emphasis appears to be placed on proof-of-safety requirements – what process, activities and documentation has to be performed/submitted to adequately demonstrate the safety of a system.

APPENDIX A

This appendix presents definitions of acronyms and terminology considered relevant to this program. The numbers in parentheses following the definitions in the Terminology section indicate the reference sources for the definitions cited. The absence of a cited reference indicates that the definition was generated specifically for this program from a variety of sources. Specific reference sources for Appendix A are listed in a glossary at the end of this appendix.

ACRONYMS

AAR - Association of American Railroads

AC - Advisory Circular

APTA - American Public Transit Association

AFISC - Air Force Inspection and Safety Center

AFSC - Air Force Systems Command

ANSI - American National Standards Institute

AREA - American Railway Engineering Association

ARES - Advanced Railroad Electronics System

ARP - Aerospace Recommended Practice

ASC - Automatic Speed Control

ASCE - American Society of Civil Engineers

ASME - American Society of Mechanical Engineers

ASQC - American Society for Quality Control

ASTM - American Society for the Testing of Materials

ATA - Air Transport Association of America

ATCS - Advanced Train Control System

BCS - British Computer Society

BIT - Built-In Test

BR - British Rail

BS - British Standard

BSI - British Standards Institution

CAD - Computer Aided Dispatching

CASE - Computer Aided Software Engineering

CATC - Continuous Automatic Train Control

CEC - Commission of the European Communities

CEN - Comité Européen de Normalisation (European Committee for Standardization)

CENELEC - Comité Européen de Normalisation Electrotechnique (European Committee for Electrotechnical Standardization)

CDR - Critical Design Review

CER - Community of European Railways

CFR - Code of Federal Regulations

CIGGT - Canadian Institute of Guided Ground Transport

COMM GRV - Commission Grande Vitesse

CPU - Central Processing Unit

CTC - Canadian Transport Commission

DB - German Federal Railways

DS - Railroad Regulations (Issued by DB)

DIN - Deutsches Institute fur Normung (German Standards Institute)

DOD - Department of Defense

DoT - Department of Transport (England)

DVS - Deutscher Verband for Schweisstechnik (German Welding Institute)

EBO - Railroad Construction and Traffic Regulations, (German)

EIA - Electronic Industries Association

EN - European Standard (Issued by European Committee for Standardization)

ERRI - European Railway Research Institute

ETCS - European Train Control System

ETSI - European Standardization Telecomms Institute

EUROCAE - European Organization for Civil Aviation Equipment

E VDI - Provisional Standard (Issued by Association of German Engineers)

EWICS - European Workshop on Industrial Computer Systems

FAA - Federal Aviation Administration

FAR - Federal Aviation Regulations

FDA - Food and Drug Administration

FRA - Federal Railroad Administration

FS - Italian Railways

FTA - Federal Transit Administration (was UMTA - Urban Mass Transportation Administration)

HSE - Health and Safety Executive

HSGGT - High Speed Guided Ground Transport

ICAO - International Civil Aviation Organization

ICE - Intercity Express

ICSE (SRS) - Interdepartmental Committee on Software Engineering Safety Related Software Working Group

IEC - International Technochemical Commission

IEE - Institute of Electrical and Engineers

IEEE - Institute of Electrical and Electronics Engineers

IRSE - Institution of Railway Signal Engineers

ISO - International Standards Organization

JAR - Joint Aviation Requirements

JPC Rail - Joint Railway Programming Committee

JNR - Japanese National Railways (Now Japan Railways Group)

JPL - Jet Propulsion Laboratory

LZB - Linienzugbeeinflussung (German continuous speed control system)

MBO - Maglev Construction and Operation Regulation, (German - Draft)

MLS - Microwave Landing System

MoD - Ministry of Defence (England)

MTBF - Mean Time Between Failure

MTBUF - Mean Time Between Unsafe Failure

MTBWF - Mean Time Between Wrongside Failure

MU - Munich (Germany)

NASA - National Aeronautics and Space Administration

NEMA - National Electric Manufacturers Association

NESC - National Electrical Safety Code

NIST - National Institute of Standards and Technology

NMI - National Maglev Initiative

NRC - Nuclear Regulatory Commission

NSWC - Naval Surface Warfare Center

PES - Programmable Electronic Systems

ORE - Office des Recherches et d'Essais (Office for Research and Experiments)

PDR - Preliminary Design Review

RAC - Railway Association of Canada

RIA - Railway Industry Association

RTCA - Radio Technical Commission for Aeronautics

RTRI - Railway Technical Research Institute

RW MSB - Regelwerk Magnetschnellbahnen--Sicherheitstechnische Anforderungen (High-Speed Maglev Trains Safety Requirements)

SAE - Society of Automotive Engineers

SEEA - Software Errors Effects Analysis

SIFT - Software Implemented Fault Tolerance

SNCF - French National Railways

SQA - Software Quality Assurance

SRE - Safety Related Electrical

SRM - Safety Related Mechanical

SSPP - System Safety Program Plan

SWHA - Software Hazard Analysis

TRB - Transportation Research Board

TGV - Train a' Grand Vitesse (High-Speed Train, French)

TÜV (Rhineland) - Technischer Uberwachungs-Verein Rheinland e.V.

UIC - Union International de Chemin de Fer (International Union of Railways)

UNIFE - Union des Industries Ferroviaires Européennes (Union of European Railway Industries)

USCOE - U. S. Army Corp of Engineers

VDE - Verband Deutscher Elektrotechniker (Association of German Electrical Technicians)

VDI - Verbands Deutscher Ingenieure (Association of German Engineers)

VDMA - Institute for Plant and Machinery Construction

VNTSC - Volpe National Transportation Systems Center

Terminology

A

Absolute Block—A block in which no train is permitted to enter while it is occupied by another train. (2)

Absolute Signal—A signal of an automatic block signal system that is capable of displaying "Stop" as opposed to "Stop and Proceed." (2)

Approach Signal—A fixed signal used in connection with one or more signals to govern the approach thereto. (2)

Acceptance Testing—Formal testing conducted to determine whether or not a system satisfies its acceptance criteria and to enable the customer to determine whether or not to accept the system. (1)

Accident—An unforeseen event or occurrence which causes death, injury, or damage to property or the environment.

Active Redundancy—That redundancy wherein all redundant items are operating simultaneously rather than being activated when needed. (11)

Algorithm—A finite set of well-defined rules for the solution of a problem in a finite number of steps. (1)

Anomaly—Deviation from nominal performance which does not cause a significant effect on system performance but does warrant investigation and/or repair. (20)

Application Software—Software designed to fulfill specific needs of a user. (1)

Architecture—The organizational structure of a system or component. (1)

Aspect (Signal Aspect)—The appearance of a fixed signal conveying an indication as viewed from the direction of an approaching train; the appearance of a cab signal conveying an indication as viewed by an observer in the cab. (2)

Assembler—A computer program that translates programs expressed in assembly language into their machine language equivalents. (1)

Assembly Language—A programming language that corresponds closely to the instruction set of a given computer, allows symbolic naming of operations and addresses, and usually results in a one-to-one translation of program instructions into machine instructions. (1)

Audit—An independent examination of a work product or set of work products to assess compliance with specifications, standards, contractual agreements, or other criteria. (1)

Automatic Block Signal (ABS) System—A series of consecutive blocks governed by block signals, cab signals, or both, actuated by a train, or engine, or by certain conditions affecting the use of a block. (2)

Automatic Train Control (ATC)—The method for automatically controlling train movement, enforcing train safety and directing train operations. (18)

Automatic Train Operation (ATO)—The portion of an ATC system that performs any or all of the functions of speed regulation, programmed stopping, door control, performance level regulation, and other functions normally assigned to a train operator. (18)

Automatic Train Protection (ATP)—The portion of an ATC system that ensures safe train movement by a combination of train detection, train separation, overspeed protection and route interlocking. (18)

Automatic Train Stop—A system in which the train is brought to a stop through automatic brake application if imposed restrictions are ignored. (20)

Automatic Train Supervision (ATS)—The portion of an ATC system that monitors system status and directs traffic movement to maintain schedules or minimize the effects of delays. (18)

Availability—The probability that a system or system element will be operational when required, expressed as the ratio of mean time between failure to the sum of mean time between failure plus mean time to restore. (18)

Axle Counter—An automatic arrangement for detecting and counting car and locomotive axles that pass a given wayside location; usually makes use of a wheel detector. (2)

B

Back-Up System—A redundant system that performs the principal functions of the primary system with minimum deviation from the performance of the primary system. (20)

Ballast Resistance—The resistance offered by the ballast, ties, etc. to the flow of leakage current from one rail of a track circuit to the other. (2)

Big-Bang Testing—A type of integration testing in which software elements, hardware elements, or both are combined all at once into an overall system, rather than in stages. (1)

Black Box Testing—Testing that ignores the internal mechanism of a system or component and focuses solely on the outputs generated in response to selected inputs and execution conditions. (1)

Block—A length of track of defined limits, the use of which by trains and engines is governed by block signals, cab signals, or both. (2)

Block Signal—A fixed signal at the entrance of a block to govern trains and engines entering and using that block. (2)

Block Signal System—A method of governing the movement of trains into or within one or more blocks by block signals or cab signals. (20)

Bottom-Up—Pertaining to an activity that starts with the lowest-level components of a hierarchy and proceeds through progressively higher levels; for example, bottom-up design; bottom-up testing. (1)

Braking Distance—The maximum distance on any portion of any railroad which any train operating on such portion of railroad at its maximum authorized speed, will travel during a full service application of the brakes, between the point where such application is initiated and the point where the train comes to a stop. (2)

Branch Testing—Testing designed to execute each outcome of each decision point in a computer program. (1)

Bubble Chart—A dataflow, data structure, or other diagram in which entities are depicted with circles (bubbles) and relationships are represented by links drawn between the circles. (1)

C

Cab Signal—A signal located in the engine control compartment or cab indicating a condition affecting the movement of train or engine and used in conjunction with interlocking signals and in conjunction with or in lieu of block signals. (2)

Central Control—That place where train control or train supervision is accomplished for the entire transit system; the train command center. (20)

Centralized Traffic Control (CTC)—A term applied to a system of railroad operation by means of which the movement of trains over routes and through blocks on a designated section of track or tracks is directed by signals controlled from a designated point without requiring the use of train orders and without the superiority of trains. (2)

Central Processing Unit (CPU)—The brain of a computing machine, usually defined by the arithmetic and logic units (ALU) plus a control section; often called a "processor," sometimes a "mainframe." (19)

Certification—A written guarantee that a system or component complies with its specified requirements and is acceptable for operational use. (1)

Channel—A path along which data passes or along which data may be stored serially. (23)

Checked Redundancy—The implementation of a function (usually safety-critical) via the use of multiple independent channels, typically having a common input and performing identical functions, in which the channel outputs are compared such that any difference/disagreement is detected (immediately or at certain intervals). A detected disagreement causes the system to revert to a safe state. (18)

Civil Speed—The maximum speed allowed in a specified section of track or guideway as determined by physical limitations of the track/guideway structure, train design, and passenger comfort. (18)

Closed Loop Braking—Braking under continuous direction of the train control system. (20)

Closed Loop Principle—The principle of control system design in which the response of a system (feedback) is continuously compared with the controlling signal to generate an error signal. (15)

Code (Rail)—The controlled pulsing of electrical energy in a line or track circuit, usually for the purpose of transmitting information. The pulses may be on/off or polarized, or both, and may also vary in duration. (2)

Code (Software)—In software engineering, computer instructions and data definitions expressed in a programming language or in a form output by an assembler, compiler, or other translator. (1)

Code Review—A meeting at which software code is presented to project personnel, managers, users, customers, or other interested parties for comment or approval. (1)

Code System—The non-vital apparatus and circuits used for forming, transmitting, receiving, and applying the codes of a supervisory control system. (2)

Coded Track Circuit—A track circuit in which the electrical energy is varied or interrupted periodically. (4)

Color Light Signal—A fixed signal in which the indications are given by the color of a light only. (4)

Command Speed (Speed Command)—The speed imposed upon a moving vehicle or train at a given point in time by the automatic train control system. (11)

Common Mode Failure—Where separate or redundant processes fail because of some event or condition which affects them all. (22)

Compiler—A computer program that translates programs expressed in a high order language into their machine language equivalents. (1)

Component Testing—Testing of individual hardware or software components or groups of related components. (1)

Computer Aided Dispatching—A term relating to the use of computers in centralized traffic control systems to aid in the dispatching of trains. (2)

Computer Instruction—A statement in a programming language, specifying an operation to be performed by a computer and the addresses or values of the associated operands; for example, Move A to B. (1)

Computer Program—A combination of computer instructions and data definitions that enable computer hardware to perform computational or control functions. (1)

Configuration Control—Ensures that any change, modification, addition, or amendment is prepared, accepted, and controlled by set procedures. (22)

Configuration Management—A process to assure that all documentation which describes a system and its various components is current and reflects the actual functional and physical characteristics of the system throughout its life cycle. (20)

Conflicting Routes—Two or more routes, opposing, converging, or intersecting, over which movements cannot be made simultaneously without possibility of collision. (2)

Consist—The makeup or composition (number and specific identity) of a train of vehicles. (14)

Constant Warning Time Device—A device used as a part of a highway grade crossing warning system to provide a relatively uniform warning time. (2)

Continuous Speed Control—A speed control concept which involves the continuous updating of the maximum allowable instantaneous train speed based on the train's current and precise location. (18)

Continuous Train Control—A type of control in which the locomotive (or engine control) apparatus is constantly in operative relation with the track elements and is immediately responsive to a change of conditions in the controlling section which affects train movement. (2)

Control Flow—The sequence in which operations are performed during the execution of a computer program. (1)

Control Flow Diagram—A diagram that depicts the set of all possible sequences in which operations may be performed during the execution of a system or program. Types include box diagram, flowchart, input-process-output chart, state diagram. (1)

Correctness—The degree to which a system or component is free from faults in its specification, design, and implementation. (1)

Critical Software—Software whose failure could have an impact on safety, or could cause large financial or social loss. (1)

Criticality—The degree of impact that a requirement, module, error, fault, failure, or other item has on the development or operation of a system. (1)

Crossover—Two turnouts with the track between the frogs arranged to form a continuous passage between two nearby and generally parallel tracks. (2)

D

Data Flow Analysis—A graphical analysis technique to trace behavior of program variables as they are initialized, modified, or referenced while the program executes. (8)

Data Flow Diagram—A diagram that depicts data sources, data sinks, data storage, and processes performed on data as nodes, and logical flow of data as links between the nodes. (1)

Data Structure—A physical or logical relationship among data elements, designed to support specific data manipulation functions. (1)

Deadman Control—A pedal or handle, or both, one of which must be kept in a depressed position while a locomotive is operating; usually the brake-valve handle and a pedal which the engineman can conveniently keep depressed at his seat. When pressure is released from both at the same time they function to cut off the power and apply the brakes. (4)

Debug—To detect, locate, and correct faults in a computer program. Techniques include use of breakpoints, desk checking, dumps, inspection, reversible execution, single-step operation, and traces. (1)

De-centralized—A control system configuration in which safety and non-safety critical functions are allocated to numerous local areas rather than confined to a single central location. (18)

Decoder—A device which transforms a received signal into a data format. (20)

Decoupling—The process of making software modules more independent of one another to decrease the impact of changes to, and errors in, the individual modules. (1)

Demodularization—In software design, the process of combining related software modules, usually to optimize system performance. (1)

Dependability—A measure of the degree to which an item is operable and capable of performing its required function at any (random) time during a specified mission profile, given item availability at the start of the mission. (7)

Design—The process of defining the architecture, components, interfaces, and other characteristics of a system or component. (1)

Design Review—A process or meeting during which a system, hardware, or software design is presented to project personnel, managers, users, customers, or other interested parties for comment or approval. Types include critical design review, preliminary design review, system design review. (1)

Desk Checking—A static analysis technique in which code listings, test results, or other documentation are visually examined, usually by the person who generated them, to identify errors, violations of development standards, or other problems. (1)

Diversity (Diverse Redundancy)—In fault tolerance, realization of the same function by different means. For example, use of different processors, storage media, programming languages, algorithms, or development teams. (1)

Door Control—Circuitry, including such safeguards and interlocks as required, which operates to open and close car doors. (20)

Down-Time—The period of time during which a system or component is not operational or has been taken out of service. (1)

Dual Channel (Computer) System—A system incorporating one or perhaps more computer(s) in each of two data paths--represents a form of hardware redundancy.

Dynamic Analysis—The process of evaluating a system or component based on its behavior during execution. (1)

Dynamic Braking—A method of braking in which the motor is used as a generator and the kinetic energy of the apparatus is employed as the actuating means of exciting a retarding force. (2)

E

Electric Locking—The combination of one or more electric locks and controlling circuits by means of which levers of an interlocking machine are locked, or the equivalent using circuits only, so that switches, signals, or other units operated in connection with signaling and interlocking, are secured against operation under certain conditions. (2)

Embedded Software—Software that is part of a larger system and performs some of the requirements of that system; for example, software used in an aircraft or rapid transit system. (1)

Emergency—A condition which could cause bodily harm or severe physical injury to persons, and/or serious damage to equipment. (11)

Emergency Braking—Irrevocable open-loop braking to a complete stop, at the maximum safe braking rate for the system (typically at a higher rate than that obtained with a service brake application). (18)

Emergency Stop—The stopping of a train by an emergency brake application which, after initiated, cannot be released until the train has stopped. (20)

Emulator—A model that accepts the same inputs and produces the same outputs as a given system. (1)

Encoder—A device that transforms the format of the supplied data into the format required for transmission. (20)

Entity Relationship Diagram—A diagram that depicts a set of real-world entities and the logical relationships among them. (1)

Error—The difference between a computed, observed, or measured value or condition and the true, specified, or theoretically correct value or condition. (1)

Error Correction—In data transmission, the process of changing one or more bits of information in a digital message to its (their) correct value.

Error Detection—In data transmission, the process of detecting one or more erroneous/invalid bits of information in a digital message.

Error Seeding—The process of intentionally adding known faults to those already in a computer program for the purpose of monitoring the rate of detection and removal, and estimating the number of faults remaining in the program. (1)

F

Fail-Operational—A characteristic design which permits continued operation in spite of the occurrence of a discrete failure. (6)

Fail-Operational Fail-Safe—A system characteristic which permits continued operation on occurrence of a failure while remaining acceptably safe. A second like failure results in the system remaining safe, but non-operational. (6)

Fail-Safe—A characteristic of a system or its elements whereby any failure or malfunction affecting safety will cause the system to revert to a state that is known to be safe. (18)

Fail-Soft—Pertaining to a system or component that continues to provide partial operational capability in the event of certain failures; for example, a traffic light that continues to alternate between red and green if the yellow light fails. (1)

Failure—The inability of a system or component to perform its required functions within specified performance requirements. (1)

Failure Analysis—The logical and systematic examination of a system to identify and analyze the probability, causes, and consequences of potential and real failure. (20)

Failure Mode—The physical or functional manifestation of a failure. For example, a system in failure mode may be characterized by slow operation, incorrect outputs, or complete termination of execution. (1)

Failure Mode and Effects Analysis (FMEA)—An inductive procedure in which potential malfunctions are identified and then analyzed as to their possible effects. (6)

Failure Mode, Effects and Criticality Analysis (FMECA)—An extension of an FMEA in which each effect is assigned a criticality index which reflects both the probability of the occurrence of the effect and the seriousness of the effect in terms of loss in performance and/or safety. (6)

Failure Rate—Rate at which failures occur as a function of time. If the failure rate is constant, it is frequently expressed as the reciprocal of mean-time-between-failure (MTBF). (20)

False Proceed (False Clear)—A failure of a system, device or appliance to indicate or function as intended which results in less restriction than is required. (2)

Fatal Error—An error that results in the complete inability of a system or component to function. (1)

Fault—A defect in a hardware device or component or an incorrect step, process, or data definition in a computer program. (1)

Fault Avoidance—Avoiding the insertion of errors into a computer program or system.

Fault Containment—Where a failure/fault in one part of a program (or system) is prevented from causing failure/faults in other parts of the system. (22)

Fault Masking—A condition in which one fault prevents the detection of another. (1)

Fault Tolerance—The built-in capability of a system to provide continued (full or limited) operation in the presence of a limited number of faults or failures. (18)

Fault Tree Analysis—An analytical technique, whereby an undesired system state is specified and the system is then analyzed in the context of its environment and operation to find all credible ways in which the undesired event could occur. (16)

Firmware—The combination of a hardware device and computer instructions and data that reside as read-only software on that device. (1)

Fixed Block (Control System)—A system control concept in which track or guideway is divided into sections of various fixed lengths, and trains are maintained at headways based in part on the presence of other trains in the various sections of track or guideway. (18)

Flow Chart—A control flow diagram in which suitably annotated geometrical figures are used to represent operations, data, or equipment, and arrows are used to indicate the sequential flow from one to another. (1)

Formal Analysis—Use of rigorous mathematical techniques to analyze the algorithms of a solution. The algorithms may be analyzed for numerical properties, efficiency, and/or correctness. (8)

Formal Testing—Testing conducted in accordance with test plans and procedures that have been reviewed and approved by a customer, user, or designated level of management. (1)

Form, Fit, and Function—In configuration management, that configuration comprising the physical and functional characteristics of an item as an entity, but not including any characteristics of the elements making up the item. (1)

Frog—A track structure used at the intersection of two running rails to provide support for wheels and passageways for their flanges, thus permitting wheels on either rail to cross the other. (2)

Full Service Braking—A non-emergency brake application which obtains the maximum brake rate consistent with the design of the primary brake system(s). (20)

Function—A defined objective or characteristic action of a system or component. (1)

Functional Specification—A document that specifies the functions that a system or component must perform. Often part of a requirements specification. (1)

Functional Testing—Testing that ignores the internal mechanism of a system or component and focuses solely on the outputs generated in response to selected inputs and execution conditions. (1)

G

Glass Box Testing—Testing that takes into account the internal mechanism of a system or component. Types include branch testing, path testing, statement testing. (1)

Grade Crossing—A crossing of highways, railroad tracks, other fixed guideways or pedestrian walks or combinations of these at the same level. (14)

Grade-Separation—A separation of intersecting streams of traffic by the provision of crossing structures or underpasses. (14)

Guideway—The surface or track, and the supporting structure, in or on which vehicles travel and which provides lateral control. (11)

H

Hamming Distance—The number of positions in two binary words of the same length with different binary characters/values.

Hard Failure—A failure that results in complete shutdown of a system. (1)

Hardware Diversity—The existence of different hardware devices (e.g., processors) in redundant channels.

Hardware Redundancy—The existence of more than one means in hardware of accomplishing a given function.

Hazard—An existing or potential condition that can result in an accident. (21)

Hazard Analysis—A systematic analysis of a system operation performed to identify hazards and make recommendations for their elimination or control during all life-cycle phases. (20)

Hazard Probability—The probability that a hazard will occur during the planned life of the system. (20)

Hazard Resolution—The analysis and subsequent actions taken to reduce, to the lowest level practical, the risk associated with an identified hazard. (20)

Hazard Severity—A qualitative measure or the worst potential consequences that could be caused by a specific hazard. (20)

Headway—The time separation between two trains traveling in the same direction on the same track, measured from the instant the head end of the leading train passes a given reference point until the head end of the train immediately following passes the same reference point. (15)

High Level Language (High Order Language)—A programming language that requires little knowledge of the computer on which a program will run, can be translated into several different machines languages, allows symbolic naming of operations and addresses, provides features designed to facilitate expression of data structures and program logic, and usually results in several machine instructions for each program statement. (1)

High-Speed—Velocity of up to 198 km/h or 125 mph.

High-Speed Rail—A rail transportation system which operates at speeds in excess of 198 km/h or 125 mph.

Highway Grade Crossing—An intersection of a highway with a railroad track at the same elevation. (2)

Highway Grade Crossing Signal—An electrically operated signal used for the warning of highway traffic at railroad-highway grade crossings. (2)

Highway Grade Crossing Warning System—An interconnection of various devices and their controls used to indicate the approach and/or presence of a train at a highway grade crossing. (2)

Home Signal—A fixed signal at the entrance of a route or block to govern trains or engines entering and using that route or block. (2)

Host Machine—A computer used to develop software intended for another computer. (1)

Hump Yard—A railroad classification yard in which the classification of cars is accomplished by pushing them over a summit, known as a hump, beyond which they run by gravity and are switched into selected tracks. (2)

I

Impedance Bond—An iron core coil of low resistance and relatively high reactance, used on electrified railroads to provide a continuous path for the return propulsion current around insulated joints and to confine the alternating current signaling energy to its own track circuit. (2)

Independent Verification and Validation (IV&V)—Verification and validation performed by an organization that is technically, managerially, and financially independent of the development organization. (1)

Informal Testing—Testing conducted in accordance with test plans and procedures that have not been reviewed and approved by a customer, user, or designated level of management. (1)

Inspection—A static analysis technique that relies on visual examination of development products to detect errors, violations of development standards, and other problems. (1)

Insulated Rail Joint—A joint in which electrical insulation is provided between adjoining rails. (2)

Integration Testing—Testing in which software components, hardware components, or both are combined and tested to evaluate the interaction between them. (1)

Interface Analysis—An analysis of module interfaces and associated variables. (16)

Interface Testing—Testing conducted to evaluate whether systems or components pass data and control correctly to one another. (1)

Interlocking—An arrangement of signals and signal appliances so interconnected that their movements must succeed each other in proper sequence and for which interlocking rules are in effect. It may be operated manually or automatically. (2)

Intermittent Fault—A temporary or unpredictable fault in a component. (1)

Intermittent (Discrete, Stepped) Speed Control—A speed control concept which involves the establishment of the maximum allowable train speed for a given section of track or guideway. (18)

Intermittent (Train) Control—A type of control in which the locomotive (or controlling car) apparatus is affected only at certain designated points, usually at signal locations. (2)

Interpreter—A computer program that translates and executes each statement or construct of a computer program before translating and executing the next. (1)

Interrupt—The suspension of a process to handle an event external to the process. (1)

L

Latent—Present and capable of becoming though not now visible or active (23).

M

Machine Code—Computer instructions and data definitions expressed in a form that can be recognized by the processing unit of a computer. (1)

Machine Language—A language that can be recognized by the processing unit of a computer. Such a language usually consists of patterns of 1's and 0's, with no symbolic naming of operations or addresses. (1)

Macro—In software engineering, a predefined sequence of computer instructions that is inserted into a program, usually during assembly or compilation, at each place that its corresponding macroinstruction appears in the program. (1)

Magnetic Levitation—Levitation of a vehicle by magnetic force; it may be either by magnetic attraction or repulsion. (11)

Maintainability—The characteristics of a system which enable it to be repaired and restored to operating condition after a component malfunction or failure; maintainability is often expressed in terms of the time to repair and restore operation. (11)

Malfunction—Any anomaly or failure wherein the system, subsystem, or component fails to function as intended. (20)

Manual Block Signal System—A block or a series of consecutive blocks, governed by block signals operated manually, upon information by telegraph, telephone or other means of communication. (2)

Manual Train Control—An operating mode in which the train responds to the actions of its operator through manipulation of the brake valve or master controller. (20)

Mean-Time-Between-Failures (MTBF)—The average time that a system or component will operate without failure or malfunction; the mean time between failures is the quotient of the operating time over the number of failures, and is a measure of reliability. (11)

Mean-Time-To-Repair (MTTR)—The expected or observed time required to repair a system or component and return it to normal operations. (1)

Memory Map—A diagram that shows where programs and data are stored in a computer's memory. (1)

Methodology—A particular procedure or set of procedures (23).

Metric (Software)—A quantitative measure of the degree to which a system, component, or process possesses a given attribute. (1)

Minimum Safe Headway—The minimum headway at which two consecutive vehicles can be operated in accordance with a specific safe stopping policy. Headways often assume that the lead vehicle cannot stop instantaneously and are determined on the basis of the maximum deceleration rate for a failed vehicle. (11)

Modularity—The degree to which a system or computer program is composed of discrete components such that a change to one component has minimal impact on other components. (1)

Modular Programming—A software development technique in which software is developed as a collection of modules. (1)

Module—A separately identified part of a computer program which performs a specific function; also a program unit that is discrete and identifiable with respect to compiling, combining with other units, and loading. (22), (1)

Motion Sensitive Device—A device used to sense the presence, motion, and direction of travel of a train. A device used to detect the movement of a train. (2)

Moving Block System—A system control concept in which the separation of trains is based upon their relative velocity and location. (18)

N

Non-vital Circuit—Any circuit the function of which does not affect the safety of train operation. (2)

Non-volatile (Memory)—Memory which does not require power to retain the stored data. (10)

N-version Programming—The independent generation of N ≥ 2 functionally equivalent software programs (called versions) from the same initial specification.

O

Object Code—Computer instructions and data definitions in a form output by an assembler or compiler. An object program is made up of object code. (1)

Open Loop—No feedback control. (20)

Operating System—A collection of software, firmware, and hardware elements that controls the execution of computer programs and provides such services as computer resource allocation, job control, input/output control, and file management in a computer system. (1)

Overspeed—In excess of maximum allowable safe command speed.

Overspeed Protection—The enforcement of existing speed limits. (18)

P

Pantograph—A current collecting apparatus having a long contact shoe which glides perpendicular to the underside of an overhead contact wire. (11)

Parse—To determine the syntactic structure of a language unit by decomposing it into more elementary subunits and establishing the relationships among the subunits. For example, to decompose blocks into statements, statements into expressions, expressions into operators and operands. (1)

Path Analysis—Analysis of a computer program to identify all possible paths through the program, to detect incomplete paths, or to discover portions of the program that are not on any path. (1)

Path Testing—Testing designed to execute all or selected paths through a computer program. (1)

Performance Specification—A document that specifies the performance characteristics that a system or component must possess. These characteristics typically include speed, accuracy, and memory usage. Often part of a requirements specification. (1)

Performance Testing—Testing conducted to evaluate the compliance of a system or component with specified performance requirements. (1)

Peripheral (Device)—A supplementary item of equipment that puts data into, or accepts data from, the computer. (19)

Permissive Block—A block in manual or controlled manual territory, based on the principle that a train other than a passenger train may be permitted to follow a train other than a passenger train in the block. (2)

Petri Net—An abstract, formal model of information flow, showing static and dynamic properties of a system. A Petri net is usually represented as a graph having two types of nodes (called places and transitions) connected by arcs, and markings (called tokens) indicating dynamic properties. (1)

Point Detector—A circuit controller which is part of the switch operating mechanism and operated by a rod connected to a switch, derail or movable point frog to indicate that the point is within a specified distance of the stock rail. (2)

Preliminary Hazard Analysis (PHA)—An analysis performed to obtain an initial risk assessment of a concept or system. (20)

Product Standard—A standard that defines what constitutes completeness and acceptability of items that are used or produced, formally or informally, during the software engineering process. (1)

Program—A combination of computer instructions and data definitions that enable computer hardware to perform computational or control functions. (1)

Programmable Read Only Memory—Memory which can both be read from and reprogrammed.

Program Stop—A train stop preceded by closed-loop braking such that the train is stopped at a designated point according to a predetermined speed-distance profile. (20)

Proof of Correctness—A formal technique used to prove mathematically that a computer program satisfies its specified requirements. (1)

Protocol—A set of conventions that govern the interaction of processes, devices, and other components within a system. (1)

Reliability—The ability of a system or component to perform its required functions under stated conditions for a specified period of time. (1)

Reliability Assessment—An analytical determination of numerical reliability of a system or portion thereof without actual demonstration testing. Such assessments usually employ mathematical modeling, use of available test results, and some use of estimated reliability figures. (20)

Requirements Analysis—The process of studying user needs to arrive at a definition of system, hardware, or software requirements. (1)

Retarder—A braking device built into a railway track to reduce the speed of cars. This can be done by means of brake shoes which, when set in position, press against the sides of the lower portion of the wheels. (2)

Risk—A measure of the severity and likelihood of an accident. (22)

Risk Analysis—The development of a quantitative estimate of risk based on engineering evaluation and mathematical techniques for combining estimates of incident consequences and frequencies. (25)

Risk Assessment—The process by which the results of a risk analysis (i.e., risk estimates) are used to make decisions, either through relative ranking of risk reduction strategies or through comparison with risk targets. (25)

Robustness—The degree to which a system or component can function correctly in the presence of invalid inputs or stressful environmental conditions. (1)

Route Integrity—The condition whereby a track/guideway section is safe for the entry and passage of a train. (18)

Route Selection (Automatic Switching for Classification Yards)—Term is applied to a desired track destination established for an individual cut of cars by operation of a push button or other selective device. (2)

Routine—A subprogram that is called by other programs and subprograms. (1)

S

Safe Stopping Distance—The maximum distance which any train, operating under worst case tolerances and conditions, will travel from the point where braking is initially requested to where the train comes to a complete stop. (18)

Safety—Freedom from danger. (18)

Safety Audit—An independent assessment of processes, activities, and documentation related to the safety assurance of specific systems or equipment.

Safety Critical—A designation placed on a system, subsystem, element, component, device, or function denoting that satisfactory operation of such is mandatory to assurance of patron, personnel, equipment, or facility safety. Such a designation dictates incorporation of special safety design features. (20)

Safety Validation—A process or set of activities performed on a completed system, software or hardware element to demonstrate compliance with safety requirements.

Safety Verification—a) An incremental confidence building process or set of activities performed following a given phase of system, software or hardware development to determine compliance with safety requirements established for that phase; b) can also be synonymous with safety validation.

Security—Freedom from intentional danger. (20)

Semantics—The relationships of symbols or groups of symbols to their meanings in a given language. (1)

Service Braking—Any non-emergency brake application of the primary braking system. (11)

Severity—The degree of impact that a requirement, module, error, fault, failure, or other item has on the development or operation of a system. (1)

Siding—An auxiliary track for meeting or passing trains. (2)

Simulator—A device, computer program, or system that behaves or operates like a given system when provided a set of controlled inputs. (1)

Single Channel (Computer) System—A system incorporating one (or perhaps more) computers, each of which performs unique functions, in a single data path.

Sneak Circuit Analysis—A procedure conducted to identify latent paths which cause occurrence of unwanted functions or inhibit desired functions assuming all components are functioning properly. (7)

Soft Failure—A failure that permits continued operation of a system with partial operational capability. (1)

Soft Tree—A term coined to describe a fault tree which is constructed on a system which includes a software interfacing with hardware. A software fault tree. (16)

Software—Computer programs, procedures, rules, and possibly associated documentation and data pertaining to the operation of a computer system. (8)

Software Diversity—A software development technique in which two or more functionally identical variants of a program are developed from the same specification by different programmers or programming teams with the intent of providing error detection, increased reliability, additional documentation, or reduced probability that programming or compiler errors will influence the end results. (1)

Software Development Cycle—The period of time that begins with the decision to develop a software product and ends when the software is delivered. This cycle typically includes a requirements phase, design phase, implementation phase, test phase, and sometimes, installation and checkout phase. (1)

Software Engineering—The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software. (1)

Software Life-Cycle—The period of time that begins when a software product is conceived and ends when the software is no longer available for use. The software life cycle typically includes a concept phase, requirements phase, design phase, implementation phase, test phase, installation and checkout phase, operation and maintenance phase, and, sometimes, retirement phase. (1)

Software Redundancy—The existence of more than one means in software of accomplishing a given function.

Software Reliability—The probability of error-free operation of a computer program for a specified period of time.

Software Tool—A computer program used in the development, testing, analysis, or maintenance of a program or its documentation. Examples include comparator, cross-reference generator, decompiler, driver, editor, flowcharter, monitor, test case generator, timing analyzer. (1)

Software Validation—The process of evaluating software during or at the end of the development process to determine whether it satisfies specific requirements.

Software Verification—The process of evaluating software to determine whether the products of a given development phase satisfy conditions imposed at the start of that phase.

Source Code—Computer instructions and data definitions expressed in a form suitable for input to an assembler. compiler, or other translator. (1)

Source Program—A computer program that must be compiled. assembled, or otherwise translated in order to be executed by a computer. (1)

Specification—A document that specifies in a complete, precise, verifiable manner, the requirements, design, behavior, or other characteristics of a system or component, and often, the procedures for determining whether these provisions have been satisfied. (1)

Speed Control—The function of adjusting the instantaneous vehicle speed to a given speed level. (11)

Speed Profile—A plot of speed against distance traveled. (20)

Standard(s)—Something established for use as a rule or basis of comparison in measuring or judging capacity, quantity, content, extent, value, quality, etc. (23)

Standard Code (of operating rules)—The operating, block signal and interlocking rules of the Association of American Railroads. (2)

Standby Redundancy—In fault tolerance, the use of redundant elements that are left inoperative until a failure occurs in a primary element. (1)

State Diagram—A diagram that depicts the states that a system or component can assume, and shows the events or circumstances that cause or result from a change from one state to another. (1)

Statement Testing—Testing designed to execute each statement of a computer program. (1)

Static Analysis—The process of evaluating a system or component based on its form, structure, content, or documentation. (1)

Stress Testing—Testing conducted to evaluate a system or component at or beyond the limits of its specified requirements. (1)

Structural Testing—Testing that takes into account the internal mechanism of a system or component. Types include branch testing, path testing, statement testing. (1)

Structured Analysis—A method for analyzing a problem and defining the requirements for a system. (25)

Structured Design—Any disciplined approach to software design that adheres to specified rules based on principles such as modularity, top-down design, and stepwise refinement of data, system structures, and processing steps. (1)

Structured Programming—Any software development technique that includes structured design and results in the development of structured programs. (1)

Subprogram—A separately compilable, executable component of a computer program. (1)

Subroutine—A routine that returns control to the program or subprogram that called it. (1)

Subsystem Hazard Analysis (SSHA)—An analysis applied to some element of the system to identify hazards associated with component failures. (20)

Super-Speed—Velocity above 317 km/h (200 mph).

Support Software—Software that aids in the development or maintenance of other software; for example, compilers, loaders, and other utilities. (1)

Switch Point—A movable tapered track rail, the point of which is designed to fit against the stock rail. (20)

Switch (Track)—A pair of switch points with their fastenings and operating rods providing the means for establishing a route from one track to another. (2)

Syntax—The structural or grammatical rules that define how the symbols in a language are to be combined to form words, phrases, expressions, and other allowable constructs. (1)

System Hazard Analysis (SHA)—An analysis performed on subsystem interfaces to determine the safety problem areas of the total system. (20)

System Integration Testing—See Integration Testing

System Life Cycle—The period of time that begins when a system is conceived and ends when the system is no longer available for use. (1)

System Safety—The application of operating, technical and management techniques and principles to the safety aspects of a system throughout its life to reduce hazards to the lowest level possible through the most effective use of available resources. (6)

System Safety Program Plan (SSPP)—A plan that describes in detail the tasks and activities of system safety management and system safety engineering required to identify, evaluate, and eliminate hazards, or reduce the risk to a level acceptable to the managing activity throughout the system life cycle. (17)

System Testing—Testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. (1)

T

Target Computer—The computer on which the software under development is intended to operate. (10)

Test Bed—An environment containing the hardware, instrumentation, simulators, software tools, and other support elements needed to conduct a test. (1)

Test Case—A set of test inputs, execution conditions, and expected results developed for a particular objective, such as to exercise a particular program path or to verify compliance with a specific requirement. (1)

Testing—The process of operating a system or component under specified conditions, observing or recording the results, and making an evaluation of some aspect of the system or component. (1)

Third Rail—An insulated electric conductor rail located alongside the running rails, from which current is collected by means of a sliding contact mechanism attached to the bogie of electric cars. (11)

Top-Down—Pertaining to an activity that starts with the highest level component of a hierarchy and proceeds through progressively lower levels; for example, top-down design; top-down testing. (1)

Traceability—The degree to which a relationship can be established between two or more products of the development process, especially products having a predecessor-successor or master-subordinate relationship to one another; for example, the degree to which the requirements and design of a given software component match. (1)

Track Circuit—An electrical circuit of which the rails of the track form a part. (2)

Track Relay—A relay receiving all or part of its operating energy through conductors of which the track rails are an essential part. (2)

Train Describer—An instrument used to give information regarding the origin, destination, class or character of trains, engines or cars moving or to be moved between given points. (2)

Train Detection—A method by which the presence of a train in a block or its more precise location is known. (20)

Transient Error—An error that occurs once, or at unpredictable intervals. (1)

Translator—A computer program that transforms a sequence of statements expressed in one language into an equivalent sequence of statements expressed in another language. (1)

Triple Modular Redundancy (TMR)—A type of redundancy in which the outputs of three or more channels are voted upon by a voter, which takes on the majority decision and latches out the disagreeing channel output; also known as two-out-of-three voting.

U

Unit Testing—Testing of individual hardware or software units or groups of related units. (1)

V

Validation—The process of evaluating a system or component during or at the end of the development process to determine whether it satisfies specified requirements. (1)

Verification—The process of evaluating a system or component to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase. (1)

Verification & Validation (V&V)—The process of determining whether the requirements for a system or component are complete and correct, the products of each development phase fulfill the requirements or conditions imposed by the previous phase, and the final system or component complies with specified requirements. (1)

Very-High Speed—Velocity in the range of 198 km/h (125 mph) to 317 km/h (200 mph).

Vital—Essential to safe train operations. (18)

Vital Circuit or Component—Any device, circuit or software module used to implement a vital function. (20)

Vital Function—A function critical to safety, performed by a system, subsystem, piece of equipment, or component.

Vital Relay—A relay, meeting certain stringent specifications, so designed that the probability of its failing to return to the prescribed state upon de-energization is so low as to be considered practically nonexistent. (11)

Volatile (Memory)—Memory that requires a continuous supply of power applied to its internal circuitry to prevent the loss of stored data. (10)

Voting—A scheme in which the outputs of three of more channels of a system implementation are compared with each other in order to determine agreement between (usually) two or more channels, and to permit continued operation in the presence of a malfunction in one of the channels. A degree of fault tolerance is thereby obtained. (18)

W

Walkthrough—A static analysis technique in which a designer or programmer leads members of the development team and other interested parties through a segment of documentation or code, and the participants ask questions and make comments about possible errors, violation of development standards, and other problems. (1)

Watch-dog (Timer)—A device (usually in hardware) which monitors a prescribed (continuous or periodic) operation of computer hardware and/or software and provides an indication when such operation has ceased.

Wayside Control—A "command and control system" whereby electronic and/or mechanical devices alongside the guideway execute all or part of the necessary decisions inherent in command and control of the vehicles. (11)

Wayside Equipment—Train control or movement apparatus which is located along the track or wayside as opposed to the control center or other remote location. (20)

Wayside Signal—A signal of fixed location along the track right-of-way. (15)

White Box Testing—Testing that takes into account the internal mechanism of a system or component. Types include branch testing, path testing, and statement testing. (1)

Y

Yard—A system of tracks within defined limits for making up trains and storing cars. (20)

Yard Speed—A speed, used within yard limits, that will permit stopping within one-half the range of vision. (20)

GLOSSARY OF REFERENCE SOURCES

1) IEEE Std 610.12-1990, Standard Glossary of Software Engineering Terminology, Institute of Electrical and Electronic Engineers, December 10, 1990.

2) Association of American Railroads Signal Manual, Section 1 - Administration. Association of American Railroads, 1991.

3) IEEE Std 1012-1986, IEEE Standard for Software Verification and Validation Plans, Institute of Electrical and Electronic Engineers, November 14, 1986.

4) Lewis, Robert G., Miller, Luther S., Welty, Gus, Ellsworth, Kenneth G., Flagg, Mason B., Railway Age's Comprehensive Railroad Dictionary, Simmons-Boardman Books Inc., 1984.

5) A Glossary of Transit Terminology, American Public Transit Association, September 1984.

6) Glossary of Reliability, Availability and Maintainability Terminology for Rail Rapid Transit, American Public Transit Association, February 1978.

7) MIL-STD-721C, Notice 1, Definitions of Terms for Reliability and Maintainability, Department of Defense, October 23, 1991.

8) FIPS PUB 101, Guideline for Lifecycle Validation, Verification, and Testing of Computer Software, U.S. Department of Commerce, National Bureau of Standards, June 6, 1983.

9) RW-MSB, High-Speed Maglev Trains; German Safety Requirements, English Translation published by FRA/VNTSC/RSPA, Report No. DOT/FRA/ORD-92/01, January 1992.

10) RTCA/DO-178A, Software Considerations on Airborne Systems and Equipment Certification, Radio Technical Commission for Aeronautics, March 1985.

11) Dictionary of Public Transport, 1st Edition, Unikon Internationale des Transports Publics (UITP), 1981.

12) Ellsworth, Kenneth G., The Car and Locomotive Cyclopedia, Fifth Edition, Simmons-Boardman Books Inc., 1984.

13) MU 8004, Principles of Technical Approval in Signalling and Communication Engineering, Section 30 050, German Federal Railroad, Federal Railroad Main Office, Munich, January 1, 1992.

14) Glossary of Urban Public Transportation Terms, Special Report 179, Transportation Research Board, National Academy of Sciences, Washington, D.C., 1978.

15) Automatic Train Control in Rail Rapid Transit, United States Congress, Office of Technology Assessment, May 1976.

16) Software System Safety Handbook, AFISC SSH 1-1, Headquarters Air Force Inspection and Safety Center, September 5, 1985.

17) MIL-STD-882B, Notice 1, System Safety Program Requirements, Department of Defense, July 1, 1987.

18) Luedeke, J., Thompson, R., Evaluation of Concepts for Safe Speed Enforcement, Battelle Final Report, April 3, 1992.

19) Edelman, Sheldon, "Glossary of Microprocessor-Based Control System Terms," Instruments and Control System, May 1979.

20) System Safety Glossary, U.S. Department of Transportation, Transportation Systems Center, June 1986.

21) MIL-STD-1574A, System Safety Program for Space and Missile Systems, Department of Defense, March 15, 1977.

22) Interim Defence Standard 00-55, (Draft), Ministry of Defence, May 1989.

23) Webster's New Collegiate Dictionary, G&C, Merriam Company, Springfield, MA, 1979.

24) Guidelines for Chemical Process Quantitative Risk Analysis, Center for Chemical Process Safety, American Institute of Chemical Engineers, New York, 1989.

25) De Marco, T., Structured Analysis and System Specification, Yourdon, New York, 1979.

APPENDIX B

This appendix contains a glossary of literature sources used for this program followed by a list of individual contacts. A glossary of reference sources can be found in Appendix A.

GLOSSARY OF LITERATURE SOURCES

1) MIL-STD-882C, "System Safety Program Requirements," U.S. Department of Defense, January 19, 1993.

2) MIL-STD-882B, Notice 1, "System Safety Program Requirements," U.S. Department of Defense, July 1, 1987.

3) DOD-STD-2167A, Revision A, "Defense System Software Development," U.S. Department of Defense, February 29, 1988.

4) MIL-STD-SDD, "Software Development and Documentation," Draft, (Revision to DOD-STD-2167A), U.S. Department of Defense, December 22, 1992.

5) DOD-STD-2168, "Defense System Software Quality Program," U.S. Department of Defense, April 29, 1988.

6) MIL-STD-1574, "System Safety Standard for Space and Missile Systems," U.S. Department of Defense, August 15, 1979.

7) MIL-STD-1629A, Procedures for Performing A Failure Mode, Effects and Criticality Analysis," U.S. Department of Defense, November 24, 1980.

8) AF Regulation 122-9, "The Nuclear Surety Design Certification Program for Nuclear Weapon System Software and Firmware," Department of the Air Force, August 24, 1987.

9) AF Regulation 122-10, "Nuclear Surety Safety Design Criteria for Nuclear Weapon Systems," Department of the Air Force, January 5, 1982.

10) DOD 5200.28.STD, "Trusted Computer Security Evaluation Criteria (TCSEC)" or "Orange Book," U.S. Department of Defense, December 1985.

11) CMU/SEI-91-TR-24, "Capability Maturity Model for Software," Research Access/U.S Department of Defense, 1991.

12) CMU/SEI-87-TR-23, "A Method for Assessing the Software Engineering Capability of Contractors," Preliminary Version, Software Engineering Institute, September 1987.

13) "TRILLIUM Model, Telecom Software Product Development Capability Assessment," Bell Canada, 1992.

14) "Software Process Improvement and Capability Determination Model (SPICE)," Project Overview, International Electrotechnical Commission (IEC) and International Standards Organization (ISO), 1993.

15) MIL-HDBK-287 "A Tailoring Guide for DOD-STD-2167A, Defense System Software Development," U.S. Department of Defense, August 11, 1989.

16) AFSC/AFLC Pamphlet 800-5, "Software Independent Verification and Validation," Department of the Air Force, May 20, 1988.

17) 14 CFR Part 21 and 25, "Certification Procedure for Products and Parts" and "Airworthiness Standards for Transport Category Airplanes," Federal Aviation Administration (FAA).

18) Advisory Circular 25.1309-1A, "System Design and Analysis," FAA, June 21, 1988.

19) RTCA/DO-178B, "Software Considerations in Airborne Systems and Equipment Certification," RTCA, December 1, 1992.

20) ARP 4754, Draft 23C, "Systems Integration Requirements," Society of Automotive Engineers (SAE), January 19, 1993.

21) ARP 4761, Draft 4, "Safety Assessment Guidelines for Civil Airborne Systems and Equipment," SAE, February 26, 1993.

22) Regulatory Guide 1.152, "Criteria for Programmable Digital Computer Systems in Safety Systems of Nuclear Power Generating Stations," Nuclear Regulatory Commission (NRC), November 1985.

23) ANSI/IEEE-ANS-7.4.3.2-1982, "Application Criteria for Programmable Digital Computer Systems in Safety Systems of Nuclear Power Generating Stations," ANSI/IEEE, July 6, 1982.

24) IEEE Standard 603-1991, "Standard Criteria for Safety Systems for Nuclear Power Generating Stations, IEEE, 1991.

25) P-7.4.3.2, Draft 7, "Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations," IEEE, 1993.

26) ASME NQA-1-1989, "Quality Assurance Program Requirements for Nuclear Facilities," ASME, 1989.

27) ASME NQA-2a-1990, Part 2.7, "Quality Assurance Requirements of Computer Software for Nuclear Facility Applications," ASME, 1990.

28) ANSI/ANS-10.4-1987, "Guidelines for the Verification and Validation of Scientific and Engineering Computer Programs for the Nuclear Industry," American Nuclear Society, May 13, 1987.

29) Wallace, D.R., Kuhn, D.R., Ippolito, L.M., and Beltracchi, L., "An Analysis of Standards for the Assurance of High Integrity Software," National Institute of Standards and Technology, U.S. Nuclear Regulatory Commission.

30) IEEE STD 467-1980, "Standard Quality Assurance Program Requirements for the Design and Manufacture of Class 1E Instrumentation and Electric Equipment for Nuclear Power Generating Stations," IEEE.

31) IEC Standard Publication 880, First Edition, "Software for Computers in the Safety Systems of Nuclear Power Stations," IEC, 1986.

32) 982C-H69002-0001, Revision 00, "Standard for Software Engineering of Safety Critical Software," Ontario Hydro/AECL-Candu, December 21, 1990.

33) "State of the Art Report on Software Important to Safety in Nuclear Power Plants," Version 2, International Atomic Energy Agency, May 13, 1993.

34) ANSI/IEEE Std 352-1987, "IEEE Guide for General Principles of Reliability Analysis of Nuclear Power Generating Station Safety Systems," ANSI/IEEE.

35) IEC/CEI 987, "Programmed Digital Computers Important to Safety for Nuclear Power Stations," IEC/CEI, November 1989.

36) ANSI/IEEE 730-1989, "Software Quality Assurance Plans, ANSI/IEEE.

37) ANSI/IEEE 828-1990, "Standard for Software Configuration Management Plans, ANSI/IEEE.

38) ANSI/IEEE 829-1983, "Standard for Software Test Documentation," ANSI/IEEE.

39) ANSI/IEEE 830-1984, "Guide for Software Requirements Specification," ANSI/IEEE.

40) ANSI/IEEE 1012-1986, "Standard for Software Verification and Validation Plans," ANSI/IEEE, February 10, 1987.

41) ANSI/IEEE 1016-1987, "Recommended Practice for Software Design Descriptions," ANSI/IEEE.

42) ANSI/IEEE 1028-1988, "Standard for Software Reviews and Audits," ANSI/IEEE.

43) "IEEE Software Engineering Standards Collection," Spring 1991 Edition, IEEE, April 5, 1991.

44) P1228, "Standard for Software Safety Plans," Draft J, February 11, 1993.

45) Bowen, J., Stavridou, V., "Safety Critical Systems, Formal Methods and Standards," Oxford University Computing Laboratory.

46) FIPS PUB 101, "Guideline for Lifecycle Validation, Verification and Testing of Computer Software," National Bureau of Standards (NBS), June 6, 1983.

47) FIPS PUB 132, "Guideline for Software Verification and Validation," NBS, 1987.

48) "A Comparison of U.S and Foreign Safety Regulations for Potential Application to Maglev Systems," Draft Final Report, Arthur D. Little, October 1992.

49) Wallace, D.R. and Fuji, R.U., "Software Verification and Validation: An Overview," IEEE Software, 1989.

50) Leveson, N.G., "Software Safety in Embedded Computer Systems," Communications of the ACM, Vol. 34, No. 2, February 1991.

51) NIST Special Publication 500-165, "Software Verification and Validation: Its Role in Computer Assurance and Its Relationship with Software Project Management Standards," NBS, 1989.

52) NHB 1700.1 (V1-B), "NASA Safety Policy and Requirements Document," National Aeronautics and Space Administration (NASA), Advance Copy, June 1993.

53) NSTS 1700.7B, "Safety Policy and Requirements for Payloads Using the National Space Transportation System," NASA.

54) "The Computer Control of Hazardous Payloads," Final Report, NASA, July 24, 1991.

55) "Draft Computer Development and Performance Requirements" for Space Shuttle Payloads, 1993.

56) "Software Safety Standard," Draft, NASA, May 26, 1993.

57) SSP 30309 (Rev. B), "Safety Analysis and Risk Assessment Requirements Document," NASA, October 1991.

58) TSS 30666, "Program Master Verification Plan: Avionics and Flight Software Integration and Verification Plan," Volume 4, Part 1, Change Request, NASA, 1993.

59) JPL D-576, "Independent Verification and Validation of Computer Software: Methodology," Jet Propulsion Lab (JPL), February 9, 1983.

60) JPL D-10058, "Software Systems Safety Handbook," JPL, May 10, 1993.

61) "A Brief Overview of NASA Langley's Research Program in Formal Methods," NASA Langley Research Center, September 18, 1992.

62) ATCS Specification 130, "Recommended Practices for Software Quality Assurance," Revision 3.0, Railway Association of Canada (RAC)/Association of American Railroads (AAR), March 1993.

63) ATCS Specification 140, "Recommended Practices for Safety and Systems Assurance," Revision 3.0, RAC/AAR, March 1993.

64) "ATCS Industry Standard Software Development Request for Information," AAR, 1993.

65) TP 10770E, "ATCS System Safety Validation Programs," Transport Canada, November 1990.

66) "FDA Policy for the Regulation of Computer Products," Food and Drug Administration (FDA), Draft, 1989.

67) "Reviewer Guidance for Computer Controlled Medical Devices Undergoing 510 (k) Review," FDA, August 29, 1991.

68) "Application of the Medical Device GMPS To Computerized Devices and Manufacturing Processes-Medical Device GMP Guidance for FDA Investigators," Draft, FDA, November 1990.

69) IEC 62 (Secretariat) 69, "Electrical Equipment in Medical Practice," Draft, IEC, March 1993.

70) IEC 65A (Secretariat) 122, "Software for Computers in the Application of Industrial Safety Related Systems," Draft, International Electrotechnical Commission (IEC), November 1991.

71) IEC 65A (Secretariat) 123, "Generic Aspects: Functional Safety of Electrical/Electronic/Programmable Electronic Safety Related Systems," Version 7, Draft, IEC, 1992.

72) DIN V VDE 0801, "Principles for Computers in Safety Related Systems," DIN/VDE, January 1990.

73) DIN V 19250, "Fundamental Safety Analyses for MSR (Measurement-Control-Regulation) Protective Devices, DIN/VDE, January 1989.

74) DIN VDE 0831, "Electrical Equipment for Railway Signalling," DIN, June 1983.

75) Mü 8004, "Principles of Technical Approval for Signalling and Communications Technology," (with supplements and revisions), German Federal Railway (DB).

76) "Minimum Requirements for Safety Related Computers in Railroad and Nuclear Engineering," Research Report, TÜV Rheinland and TÜV Deutschland, 1988.

77) Holscher, H. and Rader, J., "Microcomputers in Safety Technique-An Aid to Orientation for Developer and Manufacturer," TÜV Rheinland and TÜV Bayern, 1986.

78) "Safety Related Computers, TC7: Systems Reliability, Safety and Security," TÜV Rheinland, European Workshop on Industrial Computer Systems, 1985.

79) SBT 90.01/00/E, "Guidelines for the Assessment of Safety Relevant Computer Systems in Railroad Technology," TÜV Rheinland.

80) Krebs, H., "Verification of Safety Related Programs for a Maglev System," WP 520, TÜV Rheinland, European Workshop on Industrial Computer Systems, July 21, 1986.

81) Blomerius, J., "Status of the Safety Certification Process of the Transrapid System," TÜV Rheinland, 1993.

82) Haspel, U., "Procedure for the Coordinating Safety Certification of the New Automatic Passenger Transfer System (PTS) at the Frankfurt Rhein Main Airport," TÜV Rheinland, 1993.

83) Jopke, K., Knigge R., and Schnieder, E., "Functional Specification of Vital Computer Software for High-Speed Maglev Systems," SAFECOMP 1992.

84) Krebs, H., "Recommendations for the Determination of the Test Interval for Redundant Safety Related Systems," European Workshop on Industrial Computer Systems TC7 Safety and Security, March 1981.

85) CLC/TC9X/SC9XA/WGA1, "Railway Applications: Software for Railway Control and Protection Systems," Draft, CENELEC, 1993.

86) CLC/TC9X/SC9XA/WGA2, "Railway Applications: Safety Related Electronic Control and Protection Systems," Draft, CENELEC, April 1993.

87) TC9X-WG5B, "Dependability for Guided Transport Systems, Part 4: Specification and Demonstration of Safety," CENELEC.

88) Freudenreich, P. and Gilles, L., "Validation and Certification of the Track-To-Engine Signal Transmission System TVM 430 for the TGV-North High-Speed Train," SNCF/CSEE Transport.

89) Guilleux, B., "The Signalling of the New Lines Is Evolving Toward TVM 430," SNCF.

90) "Automatic Train Control Systems," 4.92 VT 191, Siemens AG, 1993.

91) "Chapter 3: Assessment Methods for Safety Critical Software by Siemens," Siemens AG, 1993.

92) A25000-P0001-01-0035, "Software Development Guidelines: Software for Computers in the Industrial Application of Safety Critical Systems-Methods and Tools," Siemens AG, August 12, 1992.

93) Goddard, E.O., and Zufferey, C.H., "Report of the Technical Committee, Cross-Acceptance of Vital Signalling Systems," Institution of Railway Signal Engineers (IRSE), March 12, 1992.

94) Report No. 1, "Safety System Validation With Regard to Cross-Acceptance of Signalling Systems by the Railways," IRSE, January 14, 1992.

95) FS 3019, "Safety Review Process," ABB Signal AB, January 27, 1993.

96) Sundvall, K-E., FS 2059, "Design of Fail-Safe Equipment: Organization of Safety Measures in Different Product Phases," ABB Signal AB, April 23, 1992.

97) Technical Specification No. 23:1991, "Safety Related Software for Railway Signalling," Consultive Document, Railway Industry Association (RIA), 1991.

98) Cribbens, A.H., "Solid-State Interlocking (SSI): An Integrated Electronic Signalling System for Mainline Railways," IEE Proceedings, Vol. 134, Pt. B, No. 3, May 1987.

99) Cribbens, A.H., "Microprocessors in Railway Signalling: The Solid State Interlocking," Microprocessors and Microsystems, Vol. 11, No. 5, June 1987.

100) Cribbens, A.H. and Mitchell, I.H., "The Application of Advanced Computing Techniques to the Generation and Checking of SSI Data," British Rail Research, July 23, 1991.

101) Ingleby, M. and Mitchell, I., "Proving Safety of a Railway Signalling System Incorporating Geographic Data," British Rail Research.

102) BS 5887, "Code of Practice for Testing of Computer Based Systems, British Standards Institute (BSI), 1980.

103) BS89/33006DC, "Software for Computers in the Application of Industrial Safety Related Systems," BSI.

104) BS89/33005DC, "Functional Safety of Programmable Electronic Systems: Generic Aspects," BSI.

105) 89/97714, "Guide to the Assessment of Reliability of Systems Containing Software," BSI, September 12, 1989.

106) ELS-DOC-4817 Issue A, "Code of Practice for Validation of Modifications to Previously Validated Code," British Rail Research, September 6, 1990.

107) ELS-DOC-4888 Issue A, "Code of Practice for the Validation of Safety Critical Software," British Rail Research, October 26, 1990.

108) SSU-D-SVA-RR-1, "Software Verification and Validation Policy Review," British Rail Research, November 27, 1992.

109) "Programmable Electronic Systems in Safety Related Applications, Part 1: An Introductory Guide," Health and Safety Executive, 1987.

110) "Programmable Electronic Systems in Safety Related Applications, Part 2: General Technical Guidelines, Health and Safety Executive, 1987.

111) "SafeIT, The Safety of Programmable Electronic Systems: A Government Consultation Document on Activities to Promote Safety of Computer-Controlled Systems, Part 1-Overall Approach and Part 2-Standards Framework," Interdepartmental Committee on Software Engineering (ICSE), June 1990.

112) UIC 738 R, "Processing and Transmission of Safety Information," International Union of Railways (UIC), 2nd Edition, January 1, 1990.

113) Report RP 8, "On Proving the Safety of Transmission Systems," UIC/Office of Research and Experiments (ORE), April 1986.

114) Report RP 11, "Proof of Safety of Computer Based Safety Systems," UIC/ORE, September 1987.

115) Akita, K., Watanabe, T., Hanakmura, H., and Okumura, I., "Computerized Interlocking System for Railway Signalling Control: SMILE," IEEE Transactions on Industry Applications, Vol. 1A-21, No. 4, May/June 1985.

116) Akita, K. and Nakamura, H., "Safety and Fault-Tolerance in Computer Controlled Railway Signalling Systems," International Working Conference on Dependable Computing for Critical Applications, August 23-25, 1989.

117) Guiho, G. and Hennebert, C., "SACEM Software Validation," IEEE, 1990.

118) Martin, M.J., "Vital Processing by Single Coded Unit," Matra Transport.

119) Forin, P., "Vital Coded Microprocessor Principles and Publication for Various Transit Systems," Matra Transport.

120) Abrial, J.R., "A Formal Approach to Large Software Construction," Matra Transport, March 1989.

121) NF F 71-011, "Railway Fixed Equipment and Rolling Stock, Data Processing, Software Dependability, Generalities," AFNOR, 1990.

122) NF F 71-012, "Railway Fixed Equipment and Rolling Stock, Data Processing, Software Dependability, Stresses on Software," AFNOR, 1990.

123) NF F 71-013, "Railway Fixed Equipment and Rolling Stock, Data Processing, Software Dependability, Adapted Methods for Software Safety Analyses," AFNOR, 1990.

124) ESA PSS-05-0 Issue 2, "ESA Software Engineering Standards," European Space Agency (ESA), February 1991.

125) ESA PSS-01-40 Issue 2, "System Safety Requirements for ESA Space Systems and Associated Equipment," European Space Agency.

126) STANAG 4404, "Safety Design Requirements and Guidelines for Munition Related Safety Critical Computing Systems," Draft, NATO, March 7, 1990.

127) NSWC TR 89-33, "Software Systems Safety Design Guidelines and Recommendations," Naval Surface Warfare Center, March 1989.

128) Interim Defence Standard 00-55 (Part 1)/Issue 1, "The Procurement of Safety Critical Software in Defence Equipment, Part 1: Requirements," Ministry of Defence, April 5, 1991.

129) Interim Defence Standard 00-55 (Part 2)/Issue 1, "The Procurement of Safety Critical Software in Defence Equipment, Part 2: Guidance," Ministry of Defence, April 5, 1991.

130) Interim Defence Standard 00-56 Issue 1, "Hazard Analysis and Safety Classification of the Computer and Programmable Electronic System Elements of Defence Equipment," Ministry of Defence, April 5, 1991.

131) SEB6-A, "System Safety Engineering in Software Development," EIA Bulletin, April 1990.

132) SRAS-02-S-000-3, "The SQMS Approach Applied in the Development of the S. R. ASCV Project," SASIB (no date).

133) SRAS-03-S-000-4, "Software Verification and Validation Plan for the S. R. ASCV System," SASIB (no date).

134) "V&V Methodologies for Computer Based Equipment," SASIB (no date).

135) STANAG 4452, "Safety Assessment of Munition Related Computing Systems," first draft, NATO, no date.

136) UL 1998, "Proposed First Edition of the Standard for Safety Related Software," draft, Underwriters Laboratory, July 30, 1993.

137) IS 402, "Technical Specifications for the Supply of Electronic Equipment for Safety and Signalling Systems," January 1988 Edition, Italian State Railways.

INDIVIDUAL CONTACTS

1) Mr. Chinnarao Mokkapati, Manager - Design Assurance Engineering, US&S

2) Mr. Neal Illenberg, Manager - Engineering Support, GRS

3) Mr. Hany Rizkalla, Director of Quality Assurance, ALCATEL-Canada

4) Mr. Jeff Utterbach, Manager - Product Assurance, Harmon Electronics

5) Mr. William McClaren, Chief - Current Technology Division, Transportation Development Center/Transport Canada

6) Mr. Ian Naish, Railway Safety Directorate/Transport Canada

7) Mr. Howard Moody, Manager-Advanced Train Technology, AAR

8) Mr. Robert Ayers, Manager-C(4) Systems, ARINC Research

9) Ms. Delores Wallace, National Institute of Standards and Technology/National Computer Systems Laboratory

10) Mr. Leo Beltracchi, U.S. Nuclear Regulatory Commission

11) Mr. Joseph Joyce, U.S. Nuclear Regulatory Commission

12) Mr. James Stewart, U.S. Nuclear Regulatory Commission

13) Mr. John Harauz, Senior Design Specialist-Control Computers, Ontario Hydro

14) Mr. Peter Saraceni, Jr., Program Manager-Flight Safety Research Branch, FAA

15) Mr. John Dimtroff, Acting Manager-Flight Test and Systems Branch, FAA

16) Mr. Philip White, Director of Office of Standards and Regulations, Food and Drug Administration

17) Mr. Bernard Liebler, Director-Standards and Electromedical Programs, Health Industry Manufacturers Association

18) Mr. Walter Frazier, Head of Systems Electronics Branch, NASA Headquarters

19) Ms. Kathrine Kemp, Office of Safety and Mission Assurance, NASA

20) Mr. John Kelley, Software Product Assurance Group, Jet Propulsion Laboratory

21) Ms. Karen L'Heureux, Systems Safety Office, Jet Propulsion Laboratory

22) Mr. David Tadlock, Senior Engineer, Flight Data Systems Division, NASA Houston

23) Mr. William Bates, Space Station Safety and Mission Assurance Division, Control Systems-Branch Chief, Johnson Space Center, NASA

24) Mr. George Sabolish, Software Product Assurance Manager, NASA Headquarters

25) Robert Hinson, Chief of Shuttle Data Systems Branch, Johnson Space Center/NASA

26) Mr. Jim Lloyd, Acting Safety Director, NASA Headquarters

27) Mr. Donald Sova, NASA Headquarters

28) Mr. Robert Hoii, Aerospace Engineer, Flight Systems Safety, Johnson Space Center/NASA

29) Mr. George Finelli, NASA Langley Research Center

30) Mr. Gerhard Aue, Senior Engineer, Transportation Systems Group, Siemens AG

31) Mr. Hans Knape, Engineer, Distribution Department, Transportation Systems Group, Siemens AG

32) Dr. Reder, Software Development, Transportation Systems Group, Siemens AG

33) Mr. Horst Strelow, Hardware Development, Transportation Systems Group, Siemens AG

34) Mr. Gunter Martitz, Siemens Transportation Systems (U.S.)

35) Dr. Heinrich Krebs, Institute for Software, Electronics and Railroad Technology, TÜV Rheinland

36) Mr. Joachim Blomerius, Institute for Software, Electronics and Railroad Technology, TÜV Rheinland

37) Mr. Ken Burrage, Director of Technical Standards, British Rail

38) Mr. Keith Hacker, Safety Validation Manager, British Rail

39) Mr. C.J.A. Edwards, Technical Standards Engineer, British Rail

40) Dr. Maurice Pollard, Director-Engineering Research and Development, British Rail Research

41) Mr. Michael Powell, Commercial Director, British Rail Research

42) Dr. Allen Cribbens, Head of Safety Critical Systems Unit, British Rail Research

43) Mr. R. Bell, Health and Safety Executive (U.K.)

44) Mr. Roger Short, Principal Inspecting Officer of Railways, Railway Inspectorate (U.K.)

45) Mr. Karl Lennartz, Section Head of Safety Related Systems and Safety Related Requirements, Bundesbahn Zentralamt (BZA), German Federal Railway

46) Mr. Karl-Erik Sundvall, Manager-Fail-Safe Department, ABB Signal AB

47) Mr. W.R. Smith, Deputy Director-Technical and Production, ERRI

48) Mr. Bengt Sterner, Chairman of Signalling Subcommittee for UIC; also, Swedish State Railways

49) Mr. Jacques Balause, Director of International Affairs, SNCF

50) Mr. Jean-Paul Guilloux, Chief of Signalling Department, SNCF

51) Mr. Pierre Freudenreich, Engineer-Signalling Department, SNCF

52) Ms. Nancy Gurd, Attorney, SYSTRA/SOFRERAIL

53) Mr. Jean Martin, ATC Business Development and Marketing, Matra Transport

54) Mr. Walter Schön, RAMSS Division Assistant Manager, Matra Transport

55) Mr. Jean-Louis de Montlivault, Space Activities Director, Bureau Veritas

56) Mr. Robert Record, Project Manager (Space), Bureau Veritas

57) Mr. Giuseppe Bonfigli, General Manager-Signalling Division, Sasib

58) Mr. Katsuji Akita, Chief-Signalling Laboratory, Railway Technical Research Institute

59) Mr. Yasuo Sato, General Manager, Planning Division of RTRI

60) Mr. Horoshi Tachikawa, Manager-Signal Engineering Department, Nippon signal

61) Mr. Akiyoshi Yamamoto, Deputy Director, New York Office of Japan Railways Group

62) Mr. Kazamaru Shinoya, Safety Research Laboratory, East Japan Railways

63) Hiromitsu Yoshida, Safety Research Laboratory, East Japan Railways

64) Mr. Shinichiro Asano, International Division, East Japan Railways

65) Mr. Korefumi Tashiro, Industrial System Control Section, Hitachi Research Laborato

66) Mr. Tony Zawilski, Chairman of IEEE Software Safety Working Group

67) Mr. William Brykeynski, Institute for Defense Analyses

68) Mr. Roger Fuji, Operations Manager, Systems Technology Operation, Logicon

69) Mr. Jean-Mormand Drouin, Quality Assurance, Bell Canada

70) Dr. A. Sethy, Arsenal-Federal Institute for Testing and Research (Vienna, Austria)

71) Attilio Ciancabilia, SASIB (Bologna, Italy)

7y2) Mathew Vlasaty, Engineering Team Leader, Underwriters Laboratory

Note: Special thanks to Mr. Jeff Gordon (VNTSC) and Mr. Arne Bang (FRA) for assisting the identification, procurement and/or translation of relevant documentation.

°U.S. GOVERNMENT PRINTING OFFICE: 1995