Vogtle PEmails

From: Hoellman, Jordan
Sent: Monday, October 05, 2015 3:05 PM
To: Vogtle PEmails
Subject: Design Process for AP1000 Common Q Safety Systems
Attachments: 2015-10-08 WCAP-15927 R4.pdf

Jordan Hoellman
Project Manager
NRO / DNRL / LB4
U.S. Nuclear Regulatory Commission
office: TWFN 6-F33
phone: (301) 415-5481
email: Jordan.Hoellman2@nrc.gov

Hearing Identifier: Vogtle_COL_Docs_Public
Email Number: 7

Mail Envelope Properties (06244e70bc4f4ee4b756fb8394e70972)

Subject: Design Process for AP1000 Common Q Safety Systems
Sent Date: 10/5/2015 3:04:57 PM
Received Date: 10/5/2015 3:05:01 PM
From: Hoellman, Jordan

Created By: Jordan.Hoellman2@nrc.gov

Recipients:
"Vogtle PEmails" <Vogtle.PEmails@nrc.gov>
Tracking Status: None

Post Office: HQPWMSMRS03.nrc.gov

Files	**Size**	**Date & Time**
MESSAGE	194	10/5/2015 3:05:01 PM
2015-10-08 WCAP-15927 R4.pdf	526244	

Options
Priority: Standard
Return Notification: No
Reply Requested: No
Sensitivity: Normal
Expiration Date:
Recipients Received:

WCAP-15927
APP-GW-J1R-001
Revision 4

September 2015

Design Process for AP1000 Common Q Safety Systems

WCAP-15927
APP-GW-J1R-001
Revision 4

Design Process for AP1000 Common Q Safety Systems

Matthew A. Shakun*
Product and Plant Licensing

September 2015

Verifier: Jason E. Zielinski*, Principal Engineer
AP1000 Safety Systems Software Engineering

Reviewer: Richard M. Paese*, Senior Licensing Engineer
U.S. Licensing

Reviewer: John S. Wiesemann*, Project Manager
AP1000 PMS

Approved: Mark J. Stofko*, Manager
Product and Plant Licensing

*Electronically approved records are authenticated in the electronic document management system.

Westinghouse Electric Company LLC
1000 Westinghouse Drive
Cranberry Township, PA 16066, USA

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

REVISION HISTORY

RECORD OF CHANGES

Revision	Author	Description	Completed
0	Thomas M. Hayes	Original issue.	9/18/02
1	Steven W. Gore	Class 3 DCP changes as detailed below: Added further definition of the Concept Phase (Section 1). Added additional description of life cycle (Section 1). Removed descriptions also in Common Q NRC docketed reports (Section 1). Added missing acronyms and terms (Section 2). Merged the application and platform design life cycle descriptions into one section to eliminate redundant descriptions common to both (Section 3 and throughout document). Added clarification that critical anomalies had to be completed for each phase (Section 3). Added Functional Design to System Requirements (Section 3.2). Project Master Documents now referred to as Document Index (Section 3.1). Updated Figure 3-1, "Development Process," with additional V&V methods. Updated reference document numbers (throughout document and Section 4). Removed explanation of Platform System Design Phase because it is not applicable to AP1000 PMS since it describes generic architecture (Section 4 of Rev. 0).	11/21/08
2	Warren R. Odess-Gillett	Changes are Class 3 as per NSNP 3.4.1. Updated Figure 3-1 per RAI response RAI-SRP 7.1-ICE-10, reference the SPM for the operation, maintenance and retirement software life cycle phases, and technical editing changes	6/3/09
3	Warren R. Odess-Gillett	Updated to reference the newly NRC-approved Common Q™ Topical Report (WCAP-16097-P-A, Rev. 3). Updated to reference the newly NRC-approved Software Program Manual for Common Q Systems (WCAP-16096-P-A, Rev. 4). Updated Section 3.1 to remove the term Document Index.	4/10/13
4	Matthew A. Shakun	The following change was made to address APP-GW-GEE-4380 and CAPAL 100320452: • Updated to include alternate processes to WCAP-16096-P-A, Rev. 4, "Software Program Manual for Common Q™ Systems" and WCAP-16097-P-A, Rev. 3, "Common Qualified Platform Topical Report"	See EDMS

REVISION HISTORY (cont.)

RECORD OF CHANGES (cont.)

4 (cont.)	Matthew A. Shakun	The following editorial changes were made: • Section 2.1 was updated to fix the acronym for AMPL • Sections 3 and 4 were updated to fix the title for IEEE Std. 1074-1995. • Reference 4.2.3 was deleted since it is not being cited in the document.	See EDMS

1 INTRODUCTION AND SCOPE

This document defines the process for system-level design, software design and implementation, and hardware design and implementation for the AP1000® protection and safety monitoring system development. This document supplements WCAP-16096-P-A, "Software Program Manual for Common Q™ Systems" (Reference 4.2.1). Project definition activities are described in this document as a Conceptual Phase (see Section 3.1). The Conceptual Phase is a preparatory phase before the system design begins; it is described here because it forms the management and technical baseline for the development activities.

The objective of the development process is the production of a high quality instrumentation and control (I&C) system that is to be used for the AP1000 protection and safety monitoring system. The design of the system is derived from functional and other requirements applicable to AP1000 (in addition to general requirements that may apply to all similar applications).

The functional requirements of the software are, for the most part, a direct derivation of the system functional requirements. The end product of application development is an operating I&C system, so the life cycle extends through the retirement phase (the operation, maintenance and retirement phases are sufficiently covered in Reference 4.2.1).

The Common Q™ platform consists primarily of the Asea Brown Boveri, Inc. (ABB) Advant® Controller 160 (AC160) hardware and software product line, including the Advant development tools. The development of the AC160 hardware and software and Advant tools is outside the scope of this document. The AC160 product line is developed commercially, and is qualified for use in Common Q applications by a process of commercial dedication. The commercial dedication process is defined in WCAP-16097-P-A, "Common Qualified Platform Topical Report" (Reference 4.2.2). The Common Q platform also has certain generic hardware and software modules that are developed by Westinghouse specifically for safety system applications and that are reusable for multiple systems of various types. The development of these reusable, generic modules is integrated into the life cycle process as described in this document.

2 DEFINITIONS

2.1 ACRONYMS

ABB	Asea Brown Boveri, Inc.
AC160	Part of the ABB Advant open control system family product line
AF100	Advant Fieldbus 100
AMPL	ABB Master Programming Language
CHT	Cabinet Hardware Test
CIT	Channel Integration Test
DCD	Design Control Document
DI	Document Index
EMC	Electromagnetic Compatibility
EST	Element Software Test
HSI	Human System Interface
HSL	High Speed Datalink
I&C	Instrumentation and Control
I/O	Input/Output
PMST	Processor Module Software Test
RSED	Reusable Software Element Document
RTA	Requirements Traceability Analysis
RTM	Requirements Traceability Matrix
SAT	Site Acceptance Testing
SDD	Software Design Description
SDS	System Design Specification
SIT	System Integration Test
SRS	Software Requirements Specification
SSD	System Specification Document
V&V	Verification and Validation

2.2 TERMS

Advant	An ABB open control system family product line.
Common Q	Common Qualified Platform – a safety system I&C platform as defined in WCAP-16097-P-A, "Common Qualified Platform Topical Report" (Reference 4.2.2).
Data Highway	A serial digital communications circuit that provides communications among several devices.
Datalink	A hardware link used for unidirectional or bi-directional communications between two process modules.
V&V	Verification and validation performed by an organization that is technically, managerially, and financially independent of the development organization.

3 AP1000-SPECIFIC APPLICATION DEVELOPMENT

This section defines the process that is followed in the design of the AP1000 protection and safety monitoring system and in the design and implementation of application hardware and software that are applied to AP1000. The general relationship of hardware, software, and system verification and validation (V&V) (including testing) to this development process is shown, but the details are defined by the V&V Plan.

The following phases occur in the development of the AP1000 protection and safety monitoring hardware and software:

1. Conceptual (Project Definition)
2. System Definition
3. Software Design
4. Hardware Design
5. Software Implementation
6. Hardware Implementation
7. System Integration
8. Installation

Note that testing activities are defined as part of the V&V process.

Figure 3-1 illustrates the relationship of the application development phases to each other and to the V&V process. It also shows the outputs of each phase. The activities and products of these phases are described in the remainder of Section 3. The flow of activities shown in Figure 3-1 is intended to expand on the classic "waterfall" lifecycle model. These activities may be both iterative and overlapping. In particular, because of the constraints of I&C projects, and considering the distributed character of the AP1000 I&C systems, work may commence on a given development phase before preceding phases are complete. For example, it is not necessary for the documentation of system functional requirements to be finished before software design and implementation can start on parts of the system for which the requirements have been defined. However, for a given development phase, all critical anomalies related to that phase must be resolved before the completion of that phase.

Figure 3-2 illustrates the relationship of the development phases defined in this document to the phases (or processes) defined in other documents, specifically IEEE Standard 1074-1995, "IEEE Standard for Developing Software Life Cycle Processes" (Reference 4.1.1); IEEE/EIA 12207.0-1996, "Industry Implementation of International Standard ISO/IEC 12207: 1995 (ISO/IEC 12207) Standard for Information Technology-Software Life Cycle Processes" (Reference 4.1.2); and WCAP-16096-P-A, "Software Program Manual for Common Q™ Systems" (Reference 4.2.1).

3.1 CONCEPTUAL PHASE

The major tasks of the Conceptual, or Project Definition, Phase are project management planning and project baselining.

The project execution strategy is established and documented. Resources, personnel, and organizational interfaces and dependencies are identified. Planning for schedule, costs, risk management, communication, and project closure is performed. Requisite processes are identified, and may include acquisition, supply, development, operation, and maintenance, and the supporting processes of configuration management, quality assurance, safety, verification, validation, and problem resolution.

The technical baseline is established and documented. Project baseline information typically includes:

- Definition of the scope of the development
- AP1000 Design Control Document (DCD)
- System Specification Documents (SSDs)
- Safety classification of all parts of the system included in the scope of development
- Plant documentation and databases
- Plant-wide I&C requirements
- Applicability of codes and standards, including decomposition of key codes and standards to specific requirements

3.2 SYSTEM DEFINITION PHASE

There are three main tasks in the system definition phase—system requirements analysis, system architectural design, and software requirements analysis. These three tasks overlap in their execution, and there may be considerable iteration among them. The output of this phase is a System Requirements/Functional Design document, a System Design Specification (SDS), and a Software Requirements Specification (SRS).

3.2.1 Platform Requirement Analysis

The Common Q platform is analyzed against the requirements for the AP1000 protection and safety monitoring system. Any modifications or additions to the Common Q platform are identified. These modifications or additions become first-time engineering projects that follow the same design process as described herein.

3.2.2 System Requirements Analysis/Functional Design

In this task, the project technical baseline (Section 3.1) is analyzed to specify the system requirements. This task produces the System Requirements document. Information in the System Requirements document includes system design requirements, system functional requirements (including function-related setpoints, and constants), system interface requirements, and human system interface (HSI) requirements. Detailed requirements for the interface of individual external signals and communications data are documented in an external signal database and an external communications database.

3.2.2.1 System Design Requirements

The system design requirements comprise the overall requirements and constraints for the system design, aside from the specific system functions and specific interface signals. The application System Requirements document incorporates, by reference, the platform system design requirements and identifies additions and/or exceptions that apply specifically to AP1000. The system design requirements include the following categories of requirements:

- Applicability of codes and standards, either in whole, or in part, or as guidance (which may be defined by reference to the applicability documented in the technical baseline)
- General design requirements: design basis, single failure criteria, integrity, independence, maintenance, manual capabilities, information display, access control, identification, calibration capabilities, reliability, and availability
- Hardware qualification: environmental, electromagnetic compatibility (EMC), and seismic
- Power and grounding
- External interface capabilities
- Performance requirements: time response, accuracy, and signal noise
- Test and diagnostic capabilities
- Design constraints and objectives

3.2.2.2 System Functional Requirements

The system functional requirements provide a complete definition of the sense and command features within the scope of the system (including non-safety functions, such as provision of data to the plant information system, control interlocks, information displays, etc.). They include the following categories of requirements. The requirements are provided by a combination of textual description, logic diagrams, mathematical formulas, and tables.

- Safety functions and corresponding protective actions (exact definition of the required response of the system for all design basis events)

- Non-safety-related functions (e.g., control interlocks, data to non-safety displays and systems)
- Performance requirements associated with functions (time response, accuracy)
- Setpoints and constants associated with functions (fixed value or range of adjustment, hysteresis)
- Response to failures and out-of-range conditions (internal and external)
- Functional diversity
- Signal diversity
- Separation and isolation requirements for individual functions or interfaces (e.g., assignment of signals and functions to separation divisions)
- Required auxiliary features, such as:
  - Maintenance bypass and trip logic
  - Automatic, manual, and/or continuous test capabilities
  - Maintenance functions

3.2.2.3 System Interface Requirements

The system interface requirements define the interface between the protection system being specified and the rest of the physical plant. The requirements include the following categories:

- System scope (defines what is included in the scope of supply)
- System boundaries:
  - Mechanical system (the plant process; generally, however, the actual boundary between the process and the protection system is the I&C boundary)
  - Electrical system (power and grounding)
  - I&C systems (a general description of the signal interfaces–detailed definition of all external signals is recorded in the external interface database)
  - Functional interfaces (description of the external systems with which the protection system interfaces, and identification of the parameters, controls, indications, and functions that are monitored or actuated)
- Requirements for associated equipment (e.g., time response of actuated equipment)
- Isolation requirements for external interfaces (e.g., individual requirements for Class 1E)

3.2.2.4 HSI Requirements

The HSI requirements identify all of the required operator and maintenance personnel interfaces; for example, displays, alarms, operator controls, and maintenance and test interfaces, including the associated functionality.

3.2.2.5 External Interface Database

The external interface database supplements the System Requirements document and contains two categories of information: external signal information and external communications information.

The database identifies each external physical signal received by or produced by the system. When the database is initially populated, it provides a unique identifier by which each signal can be referenced, and it defines the signal type, signal range, functional description, source or destination (by external system), and external identifier (e.g., tag number) of the signal. As the system design progresses, information is added to each signal to identify where the signal originates and terminates within the protection system, by cabinet, then, ultimately, by specific termination, including terminal identities and identity of the input/output (I/O) or communication module and point that provide the controller interface to each signal.

The database identifies each data item that the protection system receives or transmits via a data channel (datalink or data highway). The database identifies the data channel and defines, where applicable, the data type, range, functional description, update timing, and grouping with other data items. This database provides a unique identifier by which the data item can be referenced.

3.2.3 System Architectural Design

The system architectural design task identifies the major hardware and software elements of the system and their interconnections. This task produces the SDS requirements that are allocated among these items. In particular, the functional, HSI, and interface requirements are mapped to individual subsystems. System hardware requirements are identified. External signals are allocated to individual subsystems, and this information is added to the external interface database, as noted in subsection 3.2.2.5. Intrasystem signals and communications data are identified; details may be documented in an intrasystem interface database.

3.2.3.1 System Architecture

A description is given of the architecture of the protection system as a whole. Information provided includes the following, and typically will include architecture diagrams, hardware configuration diagrams, and textual descriptions of the architectural elements:

- Identification of all parts of the system, to the cabinet and subsystem level
- Interconnections among subsystems
- Assignment of power and grounding interfaces to specific cabinets or subsystems

- Definition of subsystem hardware configuration to a level of detail necessary to support software design and to identify any hardware or software that must be designed or procured (i.e., that is not part of the standard platform hardware and software)

- Evaluation of the selected architecture against the product qualification of the standard platform hardware and software

3.2.3.2 Functional Mapping

The system functions and performance requirements defined in the System Requirements document are assigned to individual subsystems. For most sense and command features (both safety and non-safety) this can be documented as a list or table of the functions that are defined in the system functional requirements (see subsection 3.2.2.2) with the subsystem assignment. If functions must be allocated to a particular processor within a subsystem because of separation requirements defined in the system functional requirements, that assignment is documented here as well. Auxiliary features, such as testing capabilities, are mapped to the architecture at a high level here.

3.2.3.3 Intrasystem Interface Database

The intrasystem interface database contains two categories of information: intrasystem signal information and intrasystem communications information.

This database identifies each physical signal that is connected between different subsystems within the protection system. The intrasystem interface database defines the signal type, signal range, functional description, and the source and destination(s) (by subsystem) and provides a unique identifier by which the signal can be referenced. Ultimately, this database also includes specific termination information, including terminal identities and identity of the I/O or communication module and point that provide the controller interface to each signal. The termination information, however, does not necessarily need to be included before hardware and software design can proceed.

The Intrasystem Interface Database also identifies each data item that the protection system receives or transmits via an intrasystem data channel (datalink or data highway). It identifies the data channel and defines, where applicable, the data type, range, functional description, update timing, and grouping with other data items. It provides a unique identifier by which the data item can be referenced.

3.2.4 Software Requirements Analysis

The software requirements analysis task completes the identification of the requirements for the software in the system. The outputs of this task are several reusable software element documents (RSEDs) and an SRS for the system-specific software. The requirements for the sense and command features typically will have been documented by the functional mapping documented in the SDS (see subsection 3.2.2.2). Any additional requirements will be identified in the SRS as defined in subsection 3.2.3.2.

3.2.4.1 Reusable Software Element Document (Summary and Requirements)

Reusable common software elements can be created for the AC160 product line in the form of type circuits and custom PC elements. A type circuit is a prearranged group of the smaller pre-existing commercially available software units (PC elements) into a larger, more complex software entity. Type circuits are not compiled code, but more like the ABB Master Programming Language (AMPL) macro definitions that can be saved individually and reused throughout one or more projects. Custom PC elements are compiled from source code and added to the library of standard PC elements available for AMPL programming. Common software elements that are type circuits or general purpose custom PC elements (new PC elements intended for common use in many different safety applications) are documented with a composite document referred to as an RSED. An RSED combines requirements, design description, and user information into a single document.

The portion of an RSED that contains the product of the software requirements analysis contains the following categories of information:

- An element (type circuit, functional unit, custom PC element) summary consisting of a general functional description of the element

- Requirements Specification:

  – Functional requirements (functions implemented, timing, accuracy)
  – I/O terminal descriptions (default values, data types, data ranges)
  – Overflow/error handling (range checking, failure modes, alarming)
  – Truth Table (outputs as a function of input combinations)

3.2.4.2 Software Requirements Specification

The high-level requirements for auxiliary features are refined into detailed requirements in the SRS. The SRS ensures that all requirements are documented for the software in each subsystem. This information may be in the System Requirements as they are mapped to subsystems and processors by the SDS (including information in the signal and communications databases). Additional information is documented as detailed requirements in the SRS. Information in the software requirements analysis includes:

- Software structure

- Software technical description

- Specific inputs and outputs, both those that are physical signals and information that is received from and supplied to human users and external data systems

- Valid input ranges

- Output ranges, if they must be specifically limited

- Required HSI formats (e.g., input screen formats, printed report formats)
- Required sequences of operations (e.g., test sequences, operator dialog sequences)
- Functional processing of the data
- Timing requirements or constraints
- Response to abnormal conditions and error recovery
- Retention, use, and initialization of previous state information, where required
- Safety and security requirements
- Design constraints (e.g., the required use of a particular programming tool or language, or the required use of particular platform software)

3.2.5 System Hardware Requirements

The system hardware requirements describe the hardware requirements needed to support the architecture of the protection system. Information provided includes the following:

- Identification of all the hardware elements used in the system, such as cabinets, panels, subassemblies, wiring, terminations and modules
- Definition of the hardware configuration needed to support the architecture of the protection system
- Cabinet power and grounding requirements
- Cabinet cooling requirements
- Cabinet labeling requirements
- Cabinet environmental requirements
- Cabinet shipping and storage requirements

3.3 SOFTWARE DESIGN PHASE

In the software design phase, the software requirements are decomposed and allocated to individual software components. The use of existing software components to implement the requirements is described within an existing RSED. New software components that must be created are identified and likewise documented within an RSED. The portion of an RSED that contains the product of the Software Design Phase contains any design information that is not obvious from the implementation (AMPL diagram or code comments).

The software design is described in Software Design Description (SDD) documents. A preliminary SDD is produced in the software design phase, while a final SDD is produced in the software implementation phase. There is an SDD generated for each processor module that executes unique code. Redundant processors that execute identical, or nearly identical, code may have a single SDD; this includes processors in separate divisions, if they have essentially identical code (implement the same functions).

The preliminary SDD contains the following categories of information:

- Decomposition of the required functions into software entities (modules, procedures, type circuits, etc.), including entity names and the reason for the existence of the entity
- Module timing and priority
- A description, where applicable, of how safety (sense and command) functions and auxiliary functions are combined (e.g., the functionality required in bistable and logic processors to implement periodic testing; local functionality required to support maintenance functions, such as calibration data changes). In typical cases, this description may be made generic and included in the "Design Constraints" section of the application SRS, or even in platform (non-project-specific) documentation; a reference to such generic information should be made where applicable.
- Identification of any generic type circuits or custom PC elements that need to be developed. These may be project-universal elements, applicable in multiple processors in a specific project, or they may be new platform software. In either case, their design and implementation follows the platform software development process.
- Where applicable, handling of software initialization, redundancy, and tracking

3.4 HARDWARE DESIGN PHASE

In the hardware design phase, the final construction configuration of the production hardware is specified. The production unit specific cabinet assembly drawings and cabinet configuration drawings are issued at this stage. These drawings contain all of the information necessary to produce the production unit hardware. The drawings include the following information:

- Cabinet layout details
- Cabinet assembly details
- Cabinet bill of materials
- Cabinet configuration details
- Cabinet termination frame details
- Cabinet internal wiring details

3.5 SOFTWARE IMPLEMENTATION PHASE

In the software implementation phase, the executable code modules are created, typically by use of the AMPL tools. (Non-AC160 subsystems require different tools.) The application modules are integrated with platform software to produce code modules that are downloaded into subsystem processors for V&V testing (described in a V&V plan). The final version of the RSED for all of the defined software components is an output of this phase. Descriptive information about the implementation is added to the preliminary SDD to produce the final SDD.

3.5.1 Final RSED

The implementation description (a printout of the AMPL diagram) is added to the RSED and a User's Guide section is added (providing the developer with adequate instruction to incorporate the common element into an application program). The complete RSED then contains the following information:

- The element summary
- The requirements specification
- Design information (as described in Section 3.3)
- Implementation (printout of AMPL diagram for the type circuits)
- Users Guide:
  - Detailed instantiation procedure (prerequisites, applicability, restrictions, signal connections)
  - Configuration/applications (database elements connections, I/O interfaces, high speed datalink [HSL] interfaces, Advant Fieldbus 100 (AF100) interfaces, default values used)

3.5.2 Final Software Definition Document

The following categories of information are added to produce the final SDD:

- Mapping of signal names used in the code to names used in the requirements documents and databases, where these differ
- Printouts of the AMPL function chart diagrams
- Any other non-obvious information that is needed to understand the software implementation and its interfaces. The intention is that this is an aid to the individuals who will verify or maintain the code. This should not repeat information that is clear to a knowledgeable individual reading the diagrams (or non-AMPL source code listings).

3.6 HARDWARE IMPLEMENTATION (ASSEMBLY) PHASE

In this phase, the construction of the production unit hardware system is completed using the drawings specified in Section 3.4.

3.7 SYSTEM INTEGRATION PHASE

In this phase, completed cabinets containing the applications software are connected together as an integrated system. Validation testing (described in the V&V plan) is performed to test system functionality that was not covered by the cabinet-level validation testing. System integration and testing may be done on appropriate portions (e.g., individual divisions) of the system or on the complete system.

3.8 INSTALLATION PHASE

The completed system is installed at the site. Site Acceptance Testing (SAT), described in the V&V plan, is performed to assure that the system has not been damaged by shipping and installation. The SAT also confirms proper operation of any interfaces that were not completely tested by the factory validation testing; e.g., interfaces to other plant systems.

3.9 ALTERNATIVE METHODS TO PROCESSES DEFINED IN WCAP-16096-P-A

Table 3-1 identifies alternatives to the processes defined in WCAP-16096-P-A, “Software Program Manual for Common Q Systems” (Reference 4.2.1).

3.10 ALTERNATIVE METHODS TO PROCESSES DEFINED IN WCAP-16097-P-A

Table 3-2 identifies alternatives to the processes defined in WCAP-16097-P-A, “Common Qualified Platform Topical Report” (Reference 4.2.2).

Table 3-1 Alternative Methods to the Common Q SPM

WCAP-16096-P-A Section	WCAP-16096-P-A Text	Alternative
Glossary of Terms: Project Quality Plan (PQP) 4.3.2.1 Initiation (Concept) Phase	A document that specifies alternatives or supplements to the Westinghouse QMS, Level 2, or Level 3 procedures as required to meet contractual requirements or quality standards other than those specified in the Westinghouse QMS. When the SPM refers to a PQP, it includes the Project Quality Plan and Project Plan defined in the Westinghouse Quality Procedures. Any alternatives to the SPM processes or additional project specific information for the SQAP, SVVP, SCMP or SOMP shall be documented and justified in the PQP.	Alternative A document that specifies alternatives or supplements to the Westinghouse QMS, Level 2, or Level 3 procedures as required to meet contractual requirements or quality standards other than those specified in the Westinghouse QMS. When the SPM refers to a PQP, it includes the Project Quality Plan and Project Plan (including the Software Development Plan) defined in the Westinghouse Quality Procedures.
4.3.1 Organization Exhibit 2-1 Design/IV&V Team Organization	The NA organization includes a Quality organization and an Engineering organization. The design team and the IV&V team are organized within the Engineering organization.	Alternative The NA organization includes a Quality organization and an Engineering organization. The design team and the IV&V team are in separate organizations at least to the Director level. See updated SPM Exhibit 2-1 Design/IV&V Team Organization following this table.
4.3.2.6 Site Installation and Checkout Phase	The preparation of the site test plan will be initiated during the requirements phase to support evaluation of requirement testability on-site.	Alternative A site test plan is developed in accordance with the overall digital I&C test strategy to support installation testing and the Initial Test Program.

Table 3-1 Alternative Methods to the Common Q SPM (cont.)

WCAP-16096-P-A Section	WCAP-16096-P-A Text	Alternative
4.6.2.10 Post Mortem Review	Suggestions for improvement and/or best practices that are identified during the Post Mortem Review should be documented via EXHIBIT 11-2 CORRECTIVE ACTIONS PROCESS.	Alternative Suggestions for improvement and/or best practices that are identified during the Post Mortem Review should be documented via the Corrective Action, Prevention and Learning (CAPAL) system. EXHIBIT 11-2 contains a screenshot of the Corrective Action Process (CAP) system. The CAP system has since been migrated to the Corrective Action, Prevention and Learning (CAPAL) system per Westinghouse Level 2 procedures.
5.5.1 Management of IV&V	The resources for performing the IV&V shall be identified in the Project Quality Plan (Reference 4) that is prepared by the Project Manager during the conception phase of the software life cycle.	Alternative The resources for performing the IV&V shall be identified in the AP1000 PMS SVVP that is prepared by the IV&V team during the conception phase of the software life cycle.
6.3.2 Configuration Change Control	Software Change Request Procedure, Step 5: Revised System Baseline: The SCR forms will be used as the basis to track all system changes and to verify that changes have been properly implemented and that documentation has been updated.	Alternative DCPs, E&DCRs, the Westinghouse Level 3 Request for Engineering Change (REC) process, and the Westinghouse Level 3 Configuration Management (CM) procedure are used as the basis to track all system changes, to verify that changes have been properly implemented, and to ensure documentation has been updated.
6.3.4 Configuration Audits and Reviews	Configuration Audits and Reviews 3. External audits by customers or regulators shall be coordinated by the EPM [Engineering Project Manager] who will schedule personnel to be available if additional support is required.	Alternative External audits by customers or regulators shall be coordinated by QA or Licensing who will schedule personnel to be available, if additional support is required.

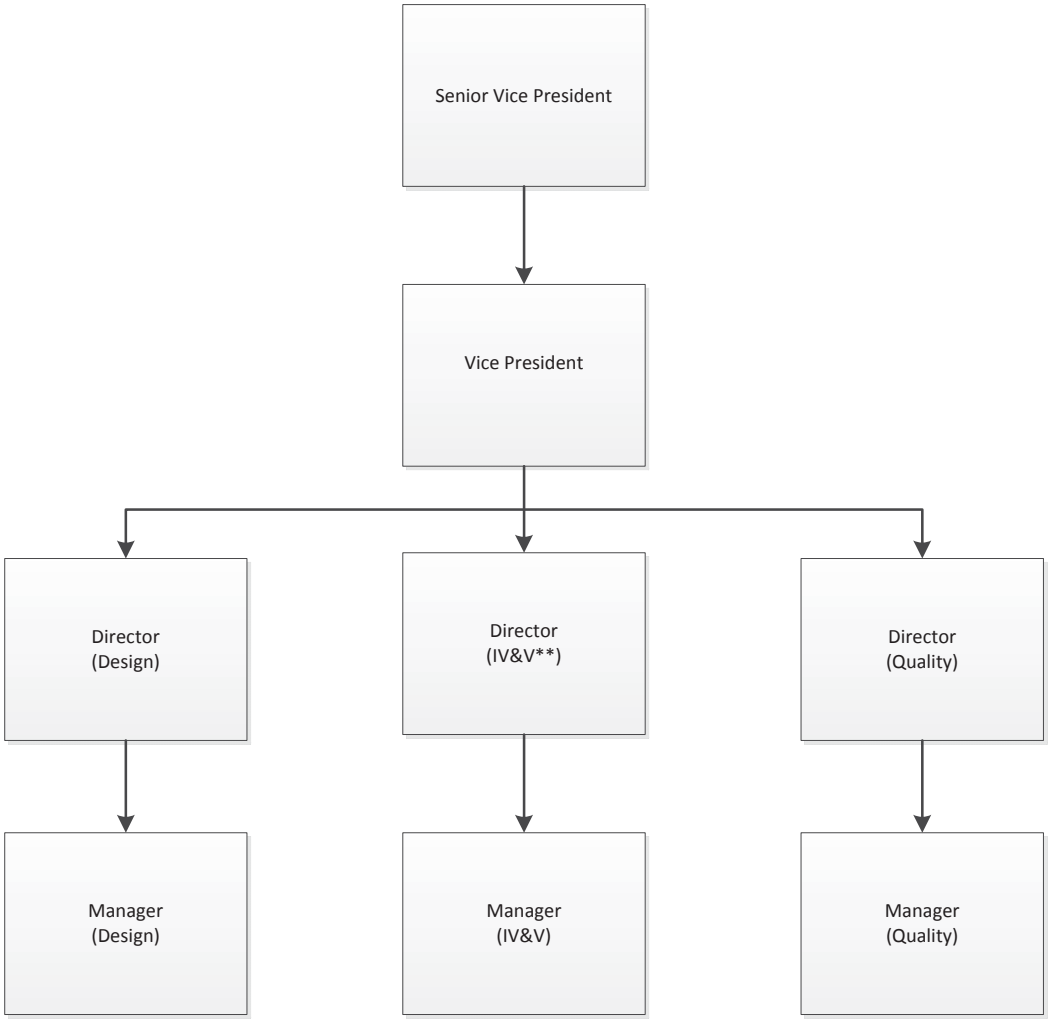
Table 3-1 Alternative Methods to the Common Q SPM (cont.)

WCAP-16096-P-A Section	WCAP-16096-P-A Text	Alternative
6.4 SCM Schedule	SCM milestones that shall be indicated on the project schedule include: • CCB establishment • Establishment of a configuration baseline, and • Implementation of change control procedures.	Alternative SCM milestones that shall be indicated in the project schedule include: • Establishment of a configuration baseline, and • Implementation of change control procedures. Establishment of the Configuration Control Board (CCB) is captured in the AP1000 I&C program plan.
9.2.3 Control	An SCR log shall be maintained for the specific Common Q™ system implementation. The Platform Lead shall confirm that the approved SCR is entered into this log.	Alternative Per the Common Q RITS Work Instruction, the RITS system maintains the SCR log. The Software Lead shall confirm that the approved SCR is entered into this log.
10.5.1 Software Verification and Validation Plan	The PQP shall also define the tracking and recording process for the hardware configuration pertinent to the software verification and validation process during all phases of the software life cycle.	Alternative The AP1000 PMS SVVP shall define the tracking and recording process for the hardware configuration (i.e., test configuration records) pertinent to the software verification and validation process during all phases of the software life cycle.
10.10 Computer Code Certificate	The completion of the implementation and checkout phase Software Verification and Validation report is the basis for the issuance of a Computer Code Certificate (see EXHIBIT 10-1 COMPUTER CODE CERTIFICATE for content requirements).	Alternative The completion of the installation and checkout phase Software Verification and Validation report is the basis for the issuance of a Computer Code Certificate (see EXHIBIT 10-1 COMPUTER CODE CERTIFICATE for content requirements).

Table 3-1 Alternative Methods to the Common Q SPM (cont.)

WCAP-16096-P-A Section	WCAP-16096-P-A Text	Alternative
11.4 Corrective Action	Corrective actions shall be documented on Exception Reports and Common Q™ Comment Records by the design team and shall be completed by the due date specified on the form…Once the independent reviewer is satisfied with the corrective action taken, the report form shall be signed.	Alternative Corrective actions shall be documented in RITS by the design team and shall be completed by the due date specified on the form…Once the RITS independent reviewer is satisfied with the corrective action taken, the report form shall be closed.
12 Secure Development and Operational Environment Plan	Secure Development and Operational Environment	Alternative The SPM, Section 12, details a Secure Development and Operational Environment Plan for Common Q systems. While this plan provides an acceptable method to comply with computer security requirements, AP1000 PMS will instead continue to use the Incorporated by Reference document APP-GW-J0R-012, "AP1000 Protection and Safety Monitoring System Computer Security Plan."

Exhibit 2-1
Westinghouse Organization Chart*



*This example organization chart shows the minimum level of separation required for the Design, IV&V, and Quality Teams

**System level validation testing is performed by another group, which meets the same level of independence as the IV&V group depicted in this organization chart

Table 3-2 Alternative Methods to the Common Q Topical Report		
WCAP-16097-P-A Section	**WCAP-16097-P-A Text**	**Alternative**
References	27. WCAP-17266, Rev. 0, "Common Q Platform Generic Change Process," Westinghouse Electric Company LLC.	Alternative 27. WCAP-17266, "Common Q Platform Generic Change Process," Westinghouse Electric Company LLC.

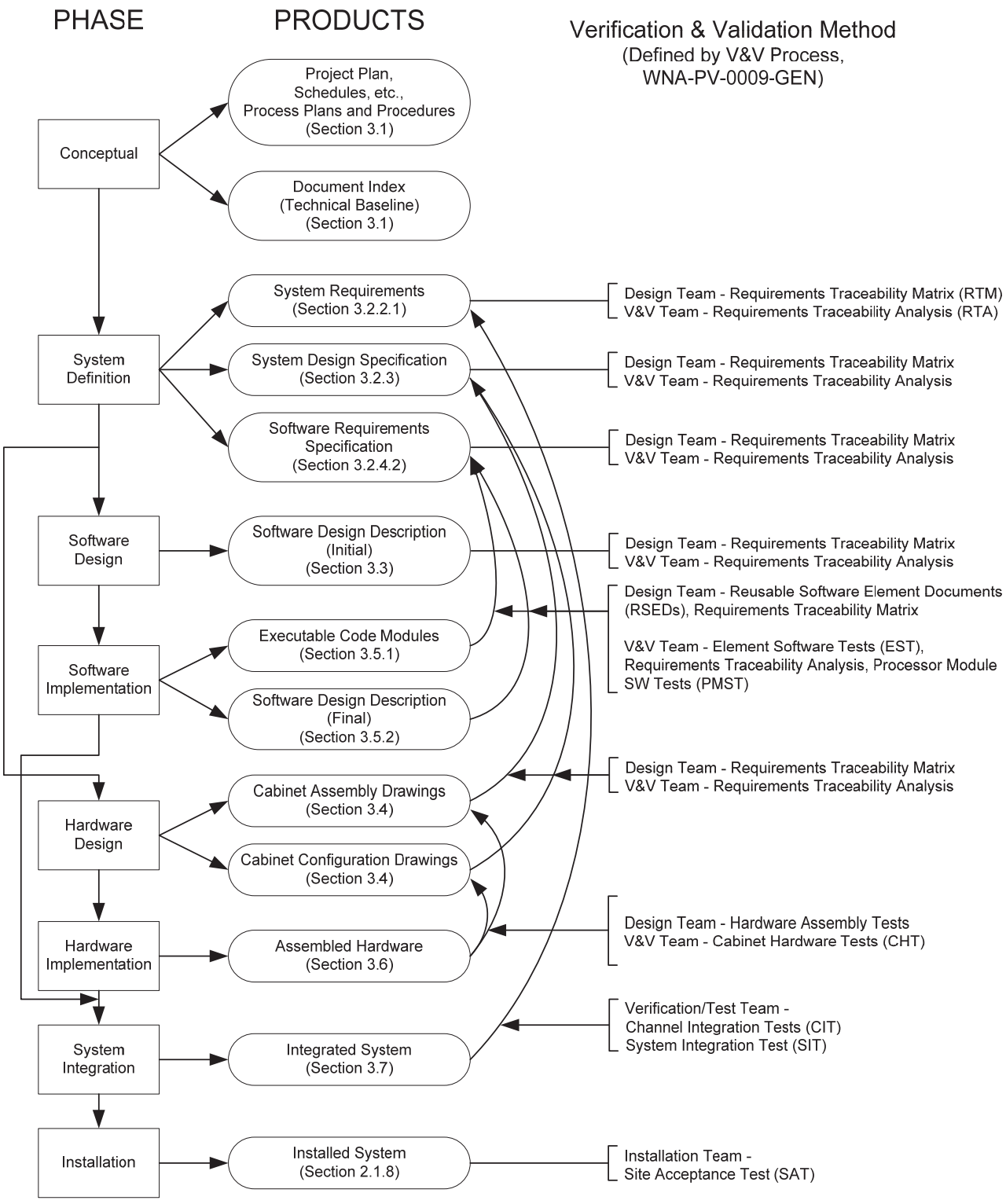


Figure 3-1 Development Process

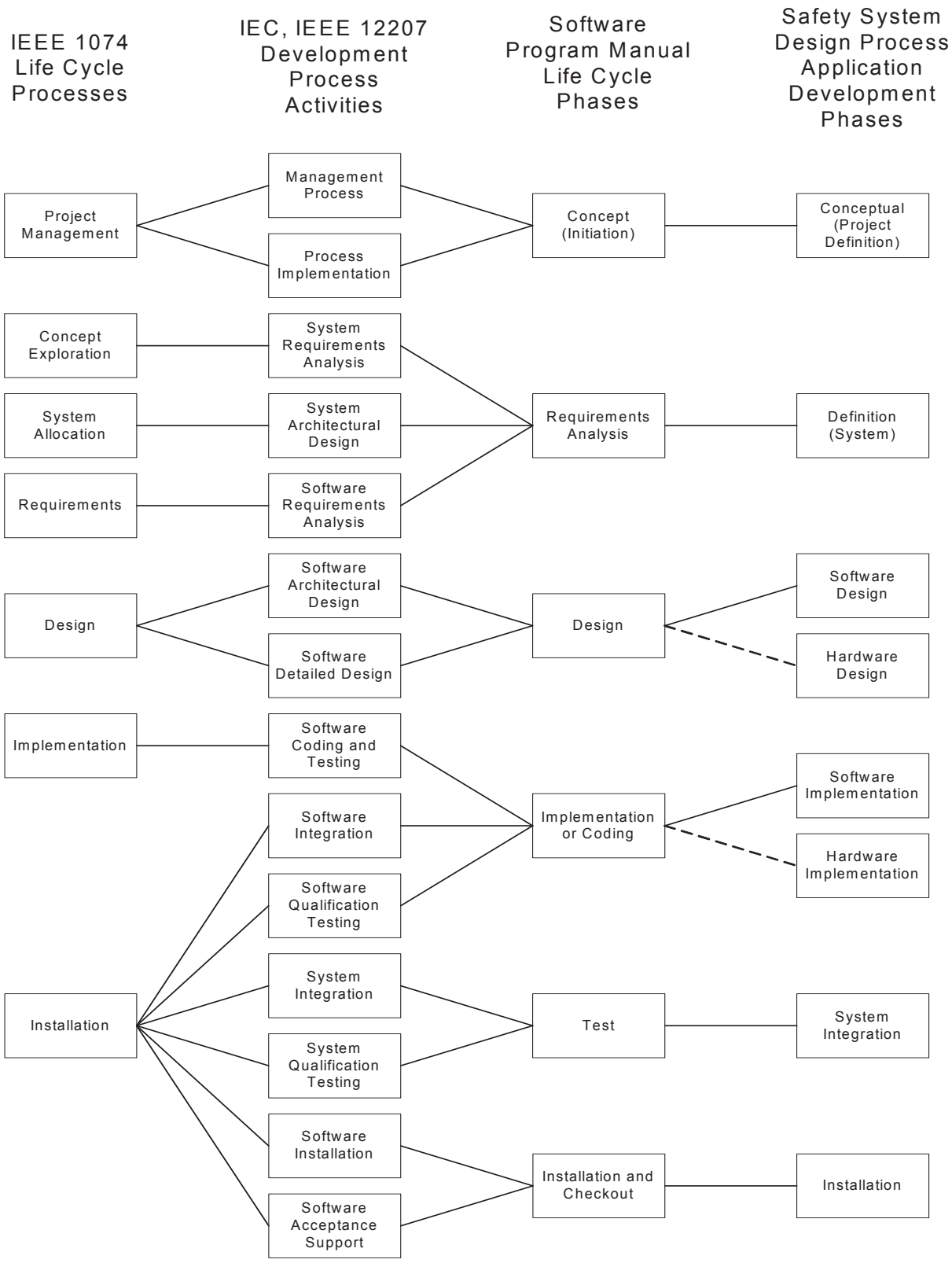


Figure 3-2 Correlation to Standard Life Cycle Phase

4 REFERENCES

4.1 INDUSTRY STANDARDS AND CODES

4.1.1 IEEE Standard 1074-1995, "IEEE Standard for Developing Software Life Cycle Processes," Institute of Electrical and Electronics Engineers, 1995.

4.1.2 IEEE/EIA 12207.0-1996, "Industry Implementation of International Standard ISO/IEC 12207: 1995 (ISO/IEC 12207) Standard for Information Technology-Software Life Cycle Processes," Institute of Electrical and Electronics Engineers/Electronic Industries Alliance, 1996.

4.2 WESTINGHOUSE DOCUMENTS

4.2.1 WCAP-16096-P-A (Proprietary), Rev. 4, "Software Program Manual for Common Q™ Systems," Westinghouse Electric Company LLC.

4.2.2 WCAP-16097-P-A (Proprietary), Rev. 3, "Common Qualified Platform Topical Report," Westinghouse Electric Company LLC.