SAFETY SOFTWARE RELEASE OPTIONS TO SUPPORT PLANT START-UP & COMMISSIONING ACTIVITIES

Jerry M. Stanley, Marc Kalo, Steve Leicher, Lori Richards
Westinghouse Electric Company LLC
5000 Ericsson Drive
Warrendale, PA 15086
stanlejm@westinghouse.com, kalomj@westinghouse.com, leichesj@westinghouse.com, richarla@westinghouse.com,

ABSTRACT

The schedule demands on new nuclear power plant site testing and commissioning programs challenge the plant's Instrumentation and Control (I&C) supplier's ability to provide certified safety software releases to meet the needs of these programs. This paper outlines the typical certified safety software development and certification process which follows a Waterfall software lifecycle model. The conflicts that result between supporting the site program and the safety software supplier's internal certification program are identified. An adapted "functional" safety release process is presented that supports the site program's needs for software changes and in parallel enables the safety software certification process to proceed. Comparisons between the two safety software release processes are examined and the benefits realized through the use of the adapted "functional" safety software release process are discussed.

Key Words: Waterfall, Software Lifecycle Model (SLM), Change Control Board (CCB), Independent Verification and Validation (IV&V)

1 INTRODUCTION

Westinghouse Electric Company has active ongoing new nuclear power plant projects that are based on the AP1000 plant design and the APR1400 plant design. For the APR1400 based projects, Westinghouse is not the plant designer but is responsible to provide the non-safety and safety system I&C systems. The Westinghouse (WEC) scope of supply on APR1400 plants includes the I&C hardware and software for both the safety and non-safety systems. The focus of this paper is to present an adaptable safety software "functional" early release process developed and used on the APR1400 United Arab Emirates (UAE) Barakah project. This process enabled the Westinghouse safety software team to be more proactive in supporting the Barakah site test team's needs to incorporate updates to the safety system software that were the result of design changes and software anomalies identified during testing.

Software developed to support products that are used in regulated industries typically requires a rigorous and highly disciplined software development lifecycle process. The nuclear power industry is highly regulated and requires that software be developed in accordance to a process that is based on industry standards and regulatory guides. The Barakah SPM defines the software lifecycle process to be utilized on the Barakah project and is based on selected industry standards and regulatory guides. The software that controls a nuclear power plant is divided into two primary subsystems, the non-safety or control software, and the safety system or primary protection software. The APR1400 UAE Barakah (UAE) safety system software follows an iterative Waterfall software development process and includes verification and validation by an independent verification and validation group (IV&V) and test group. Following the release of software by the safety software team, the IV&V and test groups perform certification activities on the safety software release. At the conclusion of the certification process a certified safety software release is issued. The certified software release can then be provided to the customer. In the early phases of new plant construction projects, the schedule for the certified software release process is not an issue. There is, however, an inherent race condition between plant construction and the availability of the certified safety system software. Typically, the certified software must be available to the site test team upon completion of construction activities in order to support the site testing and commissioning program. Once changes to the software are identified, either because of customer driven design changes or because of anomalies in the software discovered during site testing, then the Waterfall SLM is challenged to meet the demanding schedule needs of the site testing program.

On the UAE project, the Westinghouse APR1400 safety software team developed an adaptable release process in accordance with the Barakah SPM. The adapted release process enables the safety software team to be more responsive in addressing design changes and software anomalies impacting the site testing program and to mitigate schedule pressures between the ongoing certification efforts and the site testing program.

2 WESTINGHOUSE UAE BARAKAH SAFETY SYSTEM SOFTWARE DEVELOPMENT PROCESS

The Westinghouse Safety System Platform architecture consists of two key components consisting of a commercial advanced scale-able process controller and a graphics display system. The commercial advanced scale-able process controller provides I/O processing and logic operations that interact and control field components. Software programming of the process controller is performed using a graphic programming language. The graphic programming language is adept at performing logic and sequencing operations as well as arithmetic and real-time continuous control of field components. For interactions with the plant control room, the graphics display system is programmed using traditional programming languages that are hosted on “UNIX like” application servers.

Westinghouse uses a rigorous and formal software process for the development and certification of safety system software. The certified safety software is used to provide safety system protection to nuclear power plants. The software process utilizes a traditional Waterfall software life-cycle model (SLM) and references several IEEE standards that apply to the software development process. The Westinghouse safety system software team has developed a Westinghouse SPM that is applied to Westinghouse projects. This process has been approved by the NRC for use in developing safety system software for use in nuclear safety system applications. For projects where the customer provides an SPM to follow, Westinghouse performs an analysis to identify any process differences between the two SPM documents. Any differences are reconciled between the Westinghouse safety system software group and the customer.

The APR1400 UAE Barakah project software activities are governed by the Barakah project-specific SPM. The Barakah SPM and the WEC SPM share the same life-cycle phase definitions and each includes references to many of the same IEEE standards. The APR1400 UAE Barakah safety system software life-cycle phases, with a description of the work activities performed in each phase and the applicable IEEE standard reference, include:

1. Initiation or Concept Phase – This phase corresponds to the start of the software project activities. This phase is commonly referred to as the "software planning phase". In this phase the software planning documents that govern the execution of the software activities are created. These plans include:

    - Software Project Plan (SPP) [1][2] – The SPP specifies the software team organization and the roles and responsibilities of software team members and identifies the interface of the safety software team to the overall project management team, including budget tracking, software scheduling, and metrics reporting. Section 7 of the SPP includes the information necessary to define the Software Development Plan (SDP).
    - Software Quality Assurance Plan (SQAP) [3] – The SQAP describes the methodology by which all software and related documentation shall be developed and managed throughout the course of the software life-cycle activities.
    - Software Configuration Management Plan (SCMP) [4] [5] – The SCMP defines the software version control and release process and the software change control process.
    - Safety System Software Integration Plan – This plan documents the software integration strategy for integration of the safety system software with the deliverable hardware. The integration strategy includes identification of the testing required to complete the software integration testing activities.
    - Software Installation Plan (SIP) [9] [10] – The WEC SIP defines the process for transmitting software releases to site and installing the safety system software onto the target hardware.
    - Software Safety Plan (SSP) [6] – The SSP defines the activities to be performed to demonstrate the software safety requirements have been adequately identified and implemented for those safety software functions necessary to initiate reactor protection control functions. The SSP includes the process for identifying/detecting software hazards and their mitigations.
    - Safety System Software Verification and Validation Plan (SVVP) [8] – Created by the IV&V, the SVVP identifies the software verification and validation activities specific to the UAE project. Completion of the IV&V activities on a safety software release results in a certified software release.
    - Safety System Test Plan (STP) [12] – The independent test group develops the STP. The STP defines the safety system test activities to be performed by the WEC test group prior to shipment of the safety system hardware and software.

    Several project activities that impact the design groups are in process during the software concept phase. These activities include:

    - The project requirements management (RM) [7] process and configuration management (CM) process are developed and apply across the project teams.
    - System Definition Phase – For the UAE project, the customer performs the detailed plant design in accordance with plant design requirements and preliminary safety analysis reports. The WEC safety systems engineering group refines the functional and system requirements

provided by the customer in the system definition phase and creates the System Design Specification (SyDS). The SyDS forms the basis for the safety software requirements. This phase is repeated for each software release when there are changes to the system requirements.

2. Software Requirements Analysis/Software Requirements Phase – In this phase the software team creates Software Requirements Specifications (SRS). The individual requirements are linked to the parent source requirement in the SyDS. A requirements traceability matrix (RTM) is created when the SRS is finalized to make sure that all source requirements have been addressed in the software requirements phase.

   Additionally during the Software Requirements phase, the SPM includes the ability for the software team to create "software prototypes" to prove a new principle or further the functional design. When creating prototype software, the software design team is required to:

   - Follow the coding standards
   - Document the prototype design
   - Perform informal reviews by the design team
   - Provide software configuration management and software version control

3. Software Design Phase [11] – The software requirements are inputs to the software design phase. Software Design Description Documents (SDD) are created in the design phase. The SDDs provide detailed description of the software to be developed. An RTM is created upon completion of the SDDs to confirm requirement traceability from the SDD to the SRS, and from the SRS to the SyDS.
4. Software Implementation Phase or Coding Phase – The software team develops the actual software during the coding phase and performs informal code reviews and testing. At the conclusion of this phase, the software team provides a software release to the IV&V and Test teams. The release includes all the software documentation (SRSs, SDDs) and the source code. The release is documented using a Configuration Management Release Record (CMRR).
5. Test Phase – The test group is responsible for executing the test program defined in the Test Plan. This includes formal testing of the software once it has been released, installed, and integrated with the target hardware.

The phases described above document the Westinghouse APR1400 software lifecycle phases for the Barakah project. In addition to these specific software phases, the project maintains formal configuration management and requirements management processes that apply to all project design teams throughout the life cycle. The CM process includes a project Change Control Board (CCB) to evaluate and manage changes and maintain configuration control of project configuration items.

The typical safety software development and certification cycle begins with the project CCB approving a new baseline of requirements to be implemented by the design teams. The safety software team steps through each of the software phases in sequence: requirements analysis, software design, and software implementation or coding. At the conclusion of the coding phase a safety software release is provided to the IV&V and Test teams. The IV&V and Test teams use this release as their basis for testing and certification of the software with respect to a specific baseline. Once the IV&V team completes the certification of the software release, the certified software is provided to the customer. So in the typical Waterfall sequence, the certified safety software package can be used to support site testing and start-up activities. If design changes to the software are identified, then the complete process, starting with the software requirements analysis and culminating with the IV&V certification letter, is required before the revised certified software can be provided back to the customer to support the site activities. This process

is inflexible and does not allow the responsiveness that an evolving site test and start-up program requires, since the certified software is not provided to the site team until the certification process is completed.

Large complex software projects that follow a Waterfall SLM, often out of necessity, adapt the Waterfall model to become more of an iterative Waterfall. In an iterative Waterfall, a series of changes can be addressed by defining an initial change package that progresses through the sequential software lifecycle phases. As additional changes are identified, they can either be incorporated into the current release or scheduled for a future release.

The SPM includes three additional phases, the scopes of which are the responsibility of the customer. The phases are Site Installation and Checkout, Operation and Maintenance, and Retirement. The adaptive release process was developed to improve responsiveness to the site team during the Site Installation and Checkout phase activities while the WEC IV&V and Test teams performed the safety system software certification activities at the WEC home office.

3 ADAPTIVE SOFTWARE RELEASE PROCESS FOR OPTIMIZING SITE SUPPORT

The overall project schedule for the development of a nuclear power plant includes many scheduling dependencies and inherent conflicts. Inevitably, plant design changes result during the construction process. The design changes can be the result of many factors and can include: deviations between expected analytical design parameters and the actual implementation; changes in plant component suppliers; and discovery of latent design errors. These changes quite often result in a race condition between the "plant installation and checkout phase" and the availability of the I&C software development to support the checkout of the installed plant equipment. This conflict has the greatest impact on the safety system software supplier due to the Waterfall lifecycle process commonly used in safety software development as well as the rigorous IV&V and Test programs required to certify the software. It has been standard industry practice to not release software to the site program until after the safety software has been certified for use following the conclusion of the IV&V and test programs. Not only can the software lifecycle process and certification process delay the start of the initial site checkout activities, but they can create further delays as design changes and anomalies are identified by the site team or the Westinghouse IV&V and Test teams. Any identified changes will require revisions to the certified software. The safety software revisions will require additional certification activities prior to issuing a revised certified software release to the site installation and checkout team.

As the construction of the UAE Barakah plant progressed, the Westinghouse APR1400 Safety Software and IV&V teams were very much aware of the looming conflict of providing revisions to the certified software releases to support the plant site test program schedule. A number of pending plant design issues were under discussion that would drive multiple changes over time to the safety software. It was clear to the Westinghouse Barakah management team, to the customer, and to other key stakeholders that it would be difficult for the Westinghouse Barakah project team to service the growing number of future software changes and meet the aggressive site test program schedule and internal Westinghouse safety system software certification schedules. An outcome of several meetings and discussions, that included customer representation, was an adaption of the release process. The adapted release process resulted in the ability for the safety software team to support the Westinghouse safety system certification process and the ability to provide "functional" software releases under Westinghouse control to the site team in support of the site installation and checkout program. The "functional" safety software release process was developed to support the Site Installation and Checkout activities prior to loading of fuel at the plant. Once fuel has been delivered, it is imperative that safety systems are fully verified, validated, and tested.

3.1 Adaptive Barakah APR1400 Software Functional Release Process

The adapted process of providing software releases supporting the site program is based on several existing attributes of the standard safety system software development process. These include:

- The ability to provide early prototype functional software to test new or changing requirements
- An existing internal Westinghouse release process to provide "interim" releases to the Westinghouse independent test team
- An existing mature certified software release serving as the software base
- The scope and origination of the changes

The maturity of the existing UAE safety system software was a factor in the revised process. The software base was inherited from an earlier certified APR1400 software program and had already been through two extensive certification efforts on the Barakah program prior to going forward with the adapted process.

The change packages encountered and ultimately supplied to support the site program varied in size, and affect both the "functional" and certified releases. When receiving design changes, the WEC Safety Systems group and Safety Software team worked together to ensure the design input was correct. The simplest software changes required 4 to 5 working days for the safety software team to implement. Most of the software changes required 2 to 5 weeks to implement by the safety software team. Some of this time included working with the customer to confirm that the design input was correct. The IV&V and test activities were performed after the software implementation was complete and the software was released.

Many of the changes originating from the site program impacted the operation and control of field components. These types of changes are fairly localized within the I&C design and are provided on functional logic diagrams that are used by the software engineers to develop the software. Within the software, the component changes are confined to a discrete set of software routines that control the field components but also have ancillary impact on other parts of the safety system software, such as the operator displays which consist of annunciated alarms, system health displays, component status, and communications to the non-safety displays.

The Barakah site team consisted of customer personnel and Westinghouse technical support staff. When changes were identified by the customer representatives, WEC technical support staff worked to identify and agree on the scope of each change. Some of the changes were to existing implementations whereas other changes were the result of new designs. The WEC home office safety team were often consulted and provided input and review of the design changes. Once agreement on a change was reached, the change was submitted to the WEC CCB and injected into the formal software release schedule leading to certified software. The change was also scheduled for a nearer term functional release to support the site test program. Once the change was approved by the WEC CCB, the software team was able to work with the safety systems and functional design group to finalize the requirements and then implement the software changes. A set of functional releases were often grouped and scheduled to be part of the same certified software release. The ability to map the functional releases to unique certified software releases is crucial to being able to later correlate the specific functional release used in specific site testing to unique safety software certified releases. This process enabled changes to be made within one version of the safety software. Branching the software into multiple versions, which strains configuration management, was avoided.

3.2 Comparison of Certified Waterfall Process vs. Adaptive Functional Release Development

The typical Waterfall software process flow as defined in the UAE Barakah SPM is illustrated on the left side of Figure 1 and the adapted "functional" release process flow is presented on the right. The distinction between the two release methods is the "functional" releases are developed and released prior to release of the complete software package that ultimately defines the certified release. The "functional" release only includes the software code and not the software documentation which is under development while the "functional" software is being implemented and released. The typical Waterfall process that leads to certified software is sequential with the workflow moving from one phase to the next. The typical Waterfall software phases of the software lifecycle consist of Software Requirements, Software Design, and Software Implementation. At the conclusion of the Software Implementation phase, the software is released to the IV&V and Test teams to perform formal testing and certification of the software. Once the software is certified, the certified software release is provided to the customer. As shown in Figure 1, the typical Waterfall process is not well suited to providing timely releases to the site and is inflexible in providing software revisions to address site driven changes needed to support the Site Installation and Checkout phase.

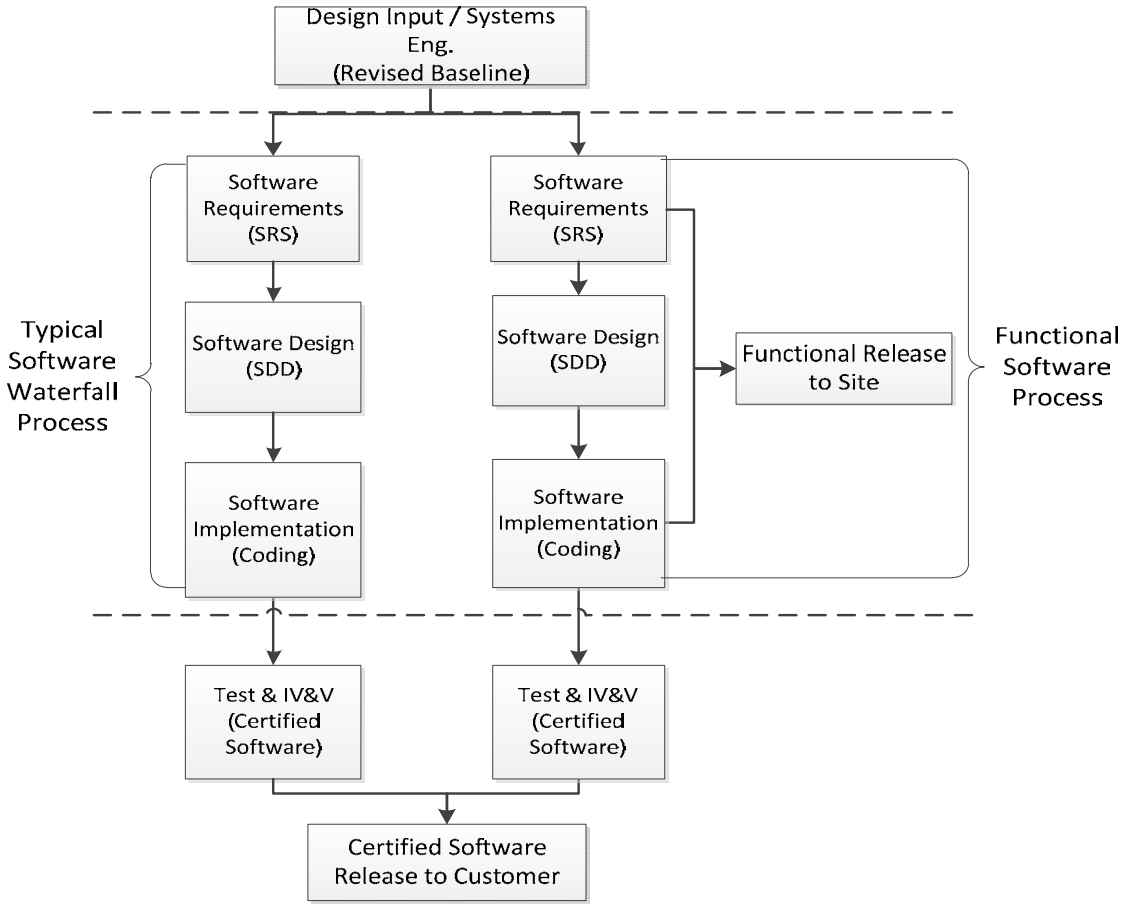

Figure 1: Typical Waterfall vs. Functional Software Release Process

The adaptive functional release process is based on a feature of the UAE Barakah SPM. The SPM includes the ability to develop prototype software to confirm the adequacy and correctness of the requirements. By providing the functional releases to the WEC site team, the operational principles of the safety system software installed onto the target plant hardware can be confirmed. The reasons mentioned earlier in Section 3.1 (the ability to provide early prototype functional software to test new or changing

requirements, an existing internal Westinghouse release process to provide "interim" releases to the Westinghouse independent test team, an existing mature certified software release serving as the software base, and the scope and origination of the changes) mitigate the risk of releasing the software prematurely. The benefits, discussed in more detail later, are obvious. The adaptive functional release process enables the ability to be proactive in meeting the needs of the site test program with controlled software releases in a relatively short time frame.

As the design team receives new changes, the system engineers and software team work together to confirm that the request is valid and properly defined. As updates to the system requirements induce a new baseline, the safety system engineers create or mark up the existing design documentation, software requirements, and make the actual coding changes. The software changes are then reviewed by the software team to confirm that the requirements documentation and coding changes are consistent with the requirements changes. The software team then performs engineering testing of the specific changes and creates the functional release. For many of the changes, the testing is performed using an additional set of plant hardware resident at the Westinghouse testing facility.

In releasing the functional software to the Westinghouse site team, the software team utilizes an existing WEC internal "Interim Software Release" process. The software team creates a release that only includes the software code, including a prototypical implementation of design changes approved per the configuration management plan. The functional releases are documented as "Interim Software Release Records", ISRR. When circumstances require additional changes, multiple functional releases to a baseline can be provided by injecting new changes to an existing baseline or by creating a revised baseline.

A disciplined change control process, especially software configuration management and version control, is a key requirement for any and all software development programs. The adaptive functional release process requires a heightened awareness of the importance of change management. In applying the adaptive functional release process, the safety software team was often managing several design packages to be included in multiple functional releases to support the site program and in the formal software release to support the safety software certification process. In order to manage the individual functional and formal software releases, key members of the safety software team were given responsibility for creating the release packages and providing them to the Westinghouse site team in accordance with the SIP. The Westinghouse site team was given responsibility to maintain and manage the installation of the software onto the plant I&C hardware while coordinating with the customer and site testing activities. A formal notification process was put in place to transfer the functional software releases from the Westinghouse safety software team to the Westinghouse site technical support team.

4 BENEFITS OF THE UAE BARAKAH APR1400 SAFETY SYSTEM SOFTWARE ADAPTIVE RELEASE PROCESS

The proposal to provide functional un-certified safety system software releases to support the Barakah site testing program was initially highly controversial within the Westinghouse safety team. However, by defining a disciplined and well managed process based on key software standard principles that include rigid design control and software change management, the program has been highly successful. The safety system software team provided 15 functional safety system software releases of varying complexity over a nine month period to support the plant start-up program. The ability to use this adaptive functional release process resulted in many benefits to Barakah project. These benefits include:

- Provided a technical solution to a commercial problem – The realization that the APR1400 safety software team would be challenged to meet the customer and plant needs to provide safety software revisions had become a significant project issue. Developing and adopting the alternative release process mitigated this issue resulting in renewed customer confidence in Westinghouse's ability to support the site test program and the overall project.

- Heightened Nuclear Safety Culture – On prior Westinghouse programs, only certified software had been used at plant installations to support site testing and startup activities. The adoption of this process created significant concerns. In order to ensure open communications and free flow of information a "Nuclear Safety Advocate" team was created. Anyone who perceived a potential safety concern could bring the concern to a member of the Nuclear Safety Advocate committee. The committee would then investigate the issue and communicate the issues to the project team. A key tenet of the "Nuclear Safety Advocate" team was that the person responsible for identifying the issue may remain anonymous. Initially, several issues were identified but as time progressed less issues were identified as people became more aware of the functional release process and comfortable that software was not being pre-maturely released without sufficient reviews and testing.
- Credit for Site Installation and Checkout Activities – The functional releases were aligned to correspond to specific baselines. Multiple functional releases may implement the changes defined by a specific baseline. The interim software releases that correspond to a specific baseline can be compared to the software released as part of a certified safety software release. By demonstrating that the software code for the functional releases is equivalent to the certified software code for a specific baseline, the site installation and checkout team can take credit for the activities performed with the functional software release.
- Improved Communications and Awareness Amongst Design Team Members – Due to the reduced cycle time of the functional releases, all members of the design team were forced to improve their communications across the team. Stakeholders supporting a change developed an attitude to take the initiative and proactively meet with one another to address changes early as well as "look out for the unexpected". Additionally, a process was put in place for changes that impacted the interfaces between the safety and non-safety teams. Prior to adopting the functional release process, the safety and non-safety teams' releases were not coordinated. With the high number of safety system functional releases, the safety and non-safety team worked out a process to notify one another when a change was identified that impacted both groups. The safety and non-safety teams were then required to include in the release documentation confirming that interface testing had been addressed. This process mitigated failures that were often not realized until miss-aligned safety and non-safety software was installed at site. The internal communication improvements further enabled better communications between the customer site and WEC site teams.
- Increased the Overall Testing of the Software – Possibly the most significant benefit of the alternative release process is the increased testing the process afforded. The additional testing occurred on the actual target system, the Nuclear Power Plant. The additional testing on the actual hardware installed at the plant identified several design and timing issues. Providing the releases to site in advance of the certified releases supports the well-known adage regarding testing, "Test early and test often".
- Configuration Management & Software Release Process – The Westinghouse design team became more committed to the change control process. The software team members and the site team members adopted a more disciplined and stricter use of software version control. The ability of the WEC site team to maintain configuration control of the safety software at site helped support troubleshooting of site specific issues. There was never a problem of not knowing what software was installed on the safety system hardware.

5 CONCLUSIONS

This paper presents a problem inherent to the safety system software development process for Nuclear Power Plants. An adaptive safety software "functional" process for releasing functional safety software is presented to mitigate the issue. The adaptive safety software functional release process supports the needs of the Site Installation and Checkout phase without compromising safety or the quality of the safety system software. The functional releases are utilized prior to fuel load.

The adaptive safety software functional release process is a highly disciplined process that relies on strict control of design changes and use of configuration management, while keeping a nuclear safety mindset. The adaptive safety software functional release process is able to meet the needs of the site testing and commissioning process all while advancing the certification of safety software releases.

6 ACKNOWLEDGMENTS

The development of the adaptive safety software functional release process was the result of input from members of the WEC safety team, especially, the APR1400 Safety System Software group, the Safety Systems Design group, and the IV&V Test groups, and Dave Jarosh. I'd especially like to thank the co-authors, Marc Kalo, Steve Leicher, and Lori Richards who provide technical software leadership to the software team and to the overall safety team.

7 REFERENCES

1. IEEE Std 1058-1998, "IEEE Standard for Software Project Management Plans".
2. IEEE Std 1074-1997, "IEEE Standard for Developing Software Life Cycle Processes".
3. IEEE Std 730-1998, "Standard for Software Quality Assurance Plans".
4. IEEE Std 828-1998, "IEEE Standard for Software Configuration Management Plans".
5. IEEE Std 1042-1987, "ANSI/IEEE Guide to Software Configuration Management" (Reaffirmed 1993).
6. IEEE Std 1228-1994, "Standard for Software Safety Plans".
7. IEEE Std 830-1998, "IEEE Recommended Practice for Software Requirements Specifications".
8. IEEE Std 1012-1998, "IEEE Standard for Software Verification and Validation".
9. IEEE Std 1063-1987, "IEEE Standard for Software User Documentation".
10. IEEE Std 1219-1998, "IEEE Standard for Software Maintenance".
11. IEEE Std 1016-1998,"IEEE Recommended Practice for Software Design Descriptions".
12. IEEE Std 829-1998, "IEEE Standard for Software and System Test Documentation".