Software Inventory Best Practices

Issued: **April 26, 2016**

Approved: Bruce F Gordon 04/26/2016

Chairperson Date

Bruce Gordon

Digitally signed by Bruce Gordon
DN: cn=Bruce Gordon, o=PVIS, ou=1231, email=bruce.gordon@aps.com, c=US
Reason: I am approving this document
Date: 2016.04.26 06:55:08 -07'00'

Team Members

Gayle Jordan, TVA, Lead

Dan Holland, PGE/DCPP

Chris Meemken, STP

Randy Moose, Duke Energy

Shanthi Palani, Exelon

Purpose

While there are requirements for nuclear power plants (utilities) to maintain an accurate inventory of software qualified for use in the nuclear environment, there is a lack of definitive guidance on what data should be included in a software inventory system. This document provides guidance for utilities in the area of software inventory which includes the software inventory data necessary for a graded approach. Software Quality Assurance (SQA) personnel can use this information as a reference when developing (or modifying) systems to maintain software application inventory data.

It is essential for utilities to maintain an inventory of software that meets and complies with defined or standardized quality specifications and thus has been approved for use for business and engineering processes. Requirements specific to the Nuclear industry can be found in 10CFR Appendix B to Part 50, IEEE Standard for Software Quality Assurance Plans Standard 730-2002,NIRMA Software Quality Assurance Documentation and Records TG16-2008, and NITSL-SQA-2005-02 Guidance Document to Implement Policy for Software Quality Assurance in the Nuclear Power Industry.

Scope

The NITSL SQA sub-committee conducted a survey in 2014 of NITSL member utilities to derive a set of core requirements for software inventory. The results of this survey indicated a commonality to data captured in a software inventory but resulted in a wide range of processes and practices used to maintain their inventory. The scope of this document is to provide guidance for utilities in the area of software inventory which includes the software inventory data necessary to track the purpose, level of importance, and current state of the application. This paper is software agnostic in that the guidance provided is independent of the type of system with which the solution is implemented. Regardless of the software used to document and maintain software inventory, certain data items must be captured and processes facilitated in the maintenance of software.

All references to software in this document refer to software, digital asset firmware, process software, and non-process software.

Definitions

Digital Asset Firmware

Digital computers, communication systems, programmable logic controller, or networks including hardware, software, and data comprised of digital devices.

Non-Process Software

Pertains to software and data used on non-plant equipment and used for plant activities. Installed and modified through Information Technology processes.

Process Software

Pertains to software installed on a safety-related system, used to control structures, systems, or components in the power plant, used for operational decisions important to the safe and reliable operation of the plant, or to meet a regulatory commitment, or for safety or quality related decisions. Installed and modified through engineering processes.

Software Developer

The individual, organization, or vendor responsible for development of or updates to a software and associated documentation including changes to the software code. The developer develops and tests the software or digital asset firmware.

Software Owner

Individual with administrative and technical responsibility for defining the functional requirements of the computer software. The software owner represents the interests of all users of the application, authorizes changes to the software, and approves the associated documentation. Also involved in user acceptance testing.

Software:

Computer programs, firmware, PLC ladder logic, procedures, rules and data pertaining to the operation of a computer system independent of the media on which it resides or the operating system it runs under. Some different types of software could include Operating System, Application, and Database Management software.

SQA Program Administrator

The person(s) with responsibility for oversight of the SQA program to ensure all software used in the nuclear environment has been properly classified, tested, documented, and all requirements are met prior to use

Software Inventory Survey Results

The NITSL SQA sub-committee member survey was conducted in 2014 to gain an understanding of the methodologies utilized by member utilities in relation to maintaining an accurate inventory of qualified software. The focus of this survey was on the tool(s) used to track and manage software information as it flows through the Software Quality Assurance (SQA) process. The survey was distributed to each NITSL member utility as well as to the SQA sub-committee membership to derive a set of core requirements for software inventory. A listing of responding utilities is shown in Attachment 4. The results of this survey indicated a commonality to methodologies of a software inventory system but resulted in a wide range of processes and practices used to maintain their software inventory. A summary of those results indicated three key commonalities of maintaining a software inventory for a nuclear utility:

- IT plays a major role in the management of software,
- Most utilities maintain SQA related software information, and
- Assessments are widely used by the utilities to monitor effectiveness of program.

Software Inventory Data, Tools, and Functionality

The focus audience for this paper is SQA Program Administrators at Nuclear Power Plants. As part of this role, the SQA PA needs to track the use of software applications at the facility and the SQA classification level of the software. This information will help determine what SQA Program requirements the Software Owner and related parties must follow.

It is recommended that the Software Owner be responsible for supplying updates as needed for the software inventory items. This is because this person has the highest level of ownership of the application and associated information.

Software Developers are very often involved with the technical aspects of the software development and/or acquisition. In these cases, the Software Owner will depend on the Software Developer to provide technical information for the inventory.
In some cases the Application Inventory system may be used by not only SQA Program Administrators but also be used by other areas of the business. For example, an Asset Management system, possibly a module of an enterprise software package, could be used for financial tracking, IT maintenance / reference, etc.

A consistent methodology should be followed to store and manage the software inventory and at a minimum should reflect the following characteristics for a top quartile inventory program:

- Software inventory data items identified
- Storage of software inventory data
- Processes to maintain software inventory data

Inventory Data

As a result of the survey of utilities in 2014, a listing of inventory data that make up a top quartile software inventory system has been assembled. The following provides guidance for inventory data that have been determined to be a core requirement of a basic software inventory system as well as additional data that provide for information accuracy.

The data items have been categorized as “Core” and “Optional.” The core items provide the required data for simple, essential functionality. Optional items provide for more advanced functionality that isn’t absolutely required to meet basic functional and regulatory requirements. The optional items should be considered when implementing the system, as they will provide for future automation and process improvement capabilities.

The table provided in Attachment 1 presents the suggested data required for a complete software inventory and their associated benefits.

Inventory Tools

A software application and associated database are a requirement for maintaining the software inventory. Many utilities already have an automated software application inventory system in place while others may have spreadsheets or a poorly designed manual process. This paper provides a starting point for those utilities that have the need to create a new software inventory system or improve on one that is not providing for the needs identified in the specified regulations or utility requirements

A computer system should be used to store the inventory of software. This system should include the following:

- Database in which to store the inventory of software
- Application interface to enter and manipulate the data

Note: In some cases, the application and database would be a single system, e.g. Microsoft Excel.

There are a variety of software tools that could provide the above functions. Attachment 2 provides a partial listing of tools that could be used to aid in the documentation and maintenance of a software inventory system.

Inventory Functionality

Best practice input has been gathered from SQA Program Administrators regarding processes that are closely related to the software inventory data. While not required, these processes provide for improved procedures and reduced iterative manual administrative tasks. It is recommended to consider these items when implementing an inventory system.

A computer system used to store the inventory of software should include functionality to ensure the data is timely and accurate. The system should also provide security ensuring authorized access only.

A partial listing of recommended functionalities of a software inventory system are shown in Attachment 3.

SQA Training

For any program or process to be successful it is important that all personnel using or interacting with the program or process be properly trained. Training should be required for all personnel with a responsibility in the SQA program. Sufficient site-wide general SQA Training is essential to provide an understanding of the SQA program and their end-user responsibilities.

SQA Assessments

Assessments of the SQA inventory are key activities to keep the program performing at a level to adequately fulfill the software management requirements. These assessments should be routinely performed and are necessary to ensure the accuracy and completeness of all phases of the software life cycle. The goals of the SQA Assessment are as follows:

- Determine if SQA requirements and processes are properly being implemented by the Software owners
- Verify that all applications are properly classified
- Information is accurate in the software inventory
- Documents as required per the classification have been prepared and are properly archived for future retrieval.

Summary

This paper has taken the next step by identifying components necessary for an efficient software inventory as well as the documentation of a set of minimum requirements for software inventory data items in support of a top-quartile SQA inventory program. Utilities should consider using this best practice information to ensure that their current software inventory systems are adequate to meet the mandates requirements related to software inventory.

As a utility identifies the need to develop or enhance their SQA program to be a top quartile program or when analyzing a software inventory system, the data fields, tools and functionality

in this paper can be used as a starting point to ensure the basic required data are captured in the effort. The utility can use the list to begin requirements conversations and to provide initial baseline items that are known to be valuable in existing utility inventory systems.

Attachments

1. Recommended Software Inventory Data
2. Application Inventory Tools
3. Functionalities of a Software Inventory System
4. Utilities That Responded to the 2014 Survey

Attachment 1

Recommended Software Inventory Data

Core Software Inventory Data Based on Graded Approach

Item	Definition/Benefits
Name	Software name or title
Description	Description of the functions the software provides.
ID Number (Unique)	The unique identifier assigned to a software application to ensure accurate identification.
SQA Plan References	Provides a link (or index ID) to the appropriate document repository where the SQA Plan is stored. Note that the SQA Plan will in turn reference additional SQA-related documents, such as test plan and requirements specification, which are outside the scope of this document.
SQA Classification Level	SQA classification denotes the capabilities of the software using a graded approach.
Basis for Classification	The justification for the SQA Classification Level
Version	Version of the software being documented, e.g. 4.2.1
Version Date	Displays the date the current version became approved for use.
Reason for Change	Description of why the software was updated, new features, security updates, product lifecycle (Automatic for hosted products etc.)
Vendor or Developer Information	The identification of the vendor the software was procured from. If developed in-house, would identify the organization or group that developed the software.
Vendor Contact Numbers	Vendor contact information for technical support, sales.
Operating System	Operating System platform and version that the software is currently running on e.g. Windows server 2012, 2008, Sun, Unix/AIX etc.
Software Owner and Organization	Identifies the person (and their reporting organization) that is responsible for the Creation and maintenance of the software inventory data. This major stakeholder is responsible for ensuring the data remains accurate. Required for SQA Plan review reminders, audits, and miscellaneous communications.
Custodian and Organization	Department or organization responsible for holding the media and performing installs.
Maintenance Contract Information	Information specific to the software maintenance contract.
Deployment Status	Identifies the current installation state of the software, e.g. Production, Test, Development, Retired or similar designations.
Software Dependencies	Listing of other applications, client operating systems, or other components that may be required to run the software e.g. Windows 7, Sun Solaris, Oracle Java 7 etc.
License Information	Number and of type of software license. How are the software licenses controlled, by License Key on server or client devices, Hardware Keys?

Optional Software Inventory Data

Item	Definitions/Benefits
Revision History	Displays historical view of previous revisions
Limitations of Use	Describes the allowed usage of the software based on the vendors license agreement.
Vendor Address	Software vendors mailing address
Database Info	Type and version of Database e.g. Oracle 11G, Microsoft SQL 2012
Installation Instructions	Link or reference to location of the software installation guide. Could also be used for additional customer specific installation instructions
Contract Number	Current vendor contract number for purchase and/or maintenance. Useful for quick lookup when working with vendor support.
References	References to related documentation and information such as Document Control numbers
Hardware Dependencies	Identification of any hardware requirements.
Hardware Configuration Environment Controls	Documentation of any hardware configurations required for the software to be installed and operational.
Workstation Type	Useful for support personnel when identifying installation or other issues. Supported software client operating systems e.g. Windows 7, 8.
Recovery Plan	Useful for quick access to the business recovery plan or document that covers how to restore the software to service when certain events may have occurred.
Software Error Notices	Useful and quick reference for verification of possible security issues or other operational errors for the listed software version.
Corrective Actions Associated	Useful for historical trending of issues and other items related to a particular version of the software.
Method of Verification on Data Input/Initial Data Load/Restrictions	Brief overview of the process, tools and data sources that may have been required to load initial data or provide data validation from other application systems, e.g. ERP, Software Asset Management etc.
Retired Date	Useful for application or system life cycle management. Date the software was removed from all installed computer systems.
App Dev Language / Architecture	Useful If the software was developed in-house or going to be maintained by internal resources, e.g. Language “C++”, Architecture “Microsoft .NET 4.0”
IT Supported (Y/N)	Useful for identification for initial point of contact for technical support. Is the software supported and maintained by your Information Technology department

Optional Software Inventory Data

(continued)

Item	Definitions/Benefits
Plant / Non-Plant	Useful in reporting for user organizations, e.g. IT and Engineering. Depending on the plant / SQA org structure, the database could be used for plant and non-plant software. The Plant and Classification values are used to determine SQA plant procedure applicability. This is done in accordance with the "graded approach" strategy.
SQA Plan Review History	Log of reviews of the SQA plan.

Attachment 2

Software Inventory Tools

General Tool Categories

Tool	Definitions/Benefits
MS Excel	Less technical end users could build and maintain databases
MS Access	Less technical end users could build and maintain databases
Application programming language and database back-end, e.g. Microsoft C# / .Net and SQL Server.	This requires more technical expertise, e.g. an IT computer programmer.
Work Management, Asset Management, or Help Desk are some examples.	Third party applications that provide a software inventory function would require less programming and depend more on proper configuration.

Note: In some cases, the application and database will be one system, e.g. Microsoft Excel.

Examples

Examples of software tools currently being used by utilities to manage software inventory:

Tool	Description of use
HP Asset Manager	The HP Asset Manager application suite is used to track software from the time of purchase until the retirement. Tracks all data necessary for a SQA software inventory. This suite provides various means to manage the inventory as well as reporting features.
BMC Remedy	Application Portfolio Management (APM) is for software inventory with basic software information including name, tracking number, technical notes, contacts and limited configuration dependencies. Remedy is part of BMC's IT Service Management product.
In-House App	Application developed with a Microsoft ASP web page front end and SQL Server database back-end.

Attachment 3

Functionality of a Software Inventory System

Functionality	Benefit
Data Lookup / Validation	For appropriate fields, data should be chosen / validated against a pick list. This helps reduce data entry errors and associated bad data.
Record Searching / Reporting	The software application / database should provide the following search capabilities: • List all software owned for a given user • Search by application name, including wildcard • List all software for a given department • Filter by plant/non-plant and classification level • Status of SQA Plans indicating review cadence • The inventory records should be selectable from the output of the searches. The output of the above should also be printable in user friendly format.
Automated Review Reminders	If a nuclear power plant has a requirement for periodic reviews of SQA Plans, an automatic process for review reminders should be considered.
Notification of personnel changes	Automated notifications of changes in personnel that have been identified as software owners should be implemented to prevent breaks in ownership of the software.

Attachment 4

NITSL Member SQA Inventory Survey

Responses were received from 16 utilities.

- Ameren – Callaway
- APS – Palo Verde
- Constellation Energy Nuclear Group
- Diablo Canyon Power Plant / PG&E
- Duke Energy
- Energy Future Holdings
- Energy Northwest
- Fermi 2
- NextEra Energy
- PPL Susquehanna
- PSEG Nuclear
- SCE & G
- Southern
- SRNS
- TVA
- Wolf Creek Nuclear Operating Corporation