TABLE OF CONTENTS

1.0 PURPOSE AND SCOPE 2
2.0 IMPLEMENTATION 3
3.0 RESPONSIBILITIES 3
4.0 PROCEDURE 4
4.1 Application of the Graded Approach 4
4.2 Initial Software Project Management and Quality Planning 5
4.3 Development and Use of Safety Spreadsheets 8
4.4 Development and Use of Non-Safety Spreadsheets 22
4.5 Software Retirement of Multiple-Use Spreadsheets 33
4.6 Use of this Procedure by Subcontractors 33
5.0 DEFINITIONS 35
6.0 RECORDS 37
7.0 SOURCES 37
7.1 Requirements 37
7.2 References 37

TABLE OF ATTACHMENTS

ATTACHMENT A - SPREADSHEET DOCUMENTATION REQUIREMENTS 39
ATTACHMENT B - DEVELOPING SPREADSHEET REQUIREMENTS 40
ATTACHMENT C - GOOD SPREADSHEET DESIGN PRACTICES 42
ATTACHMENT D - SPREADSHEET HINTS AND TIPS 45
ATTACHMENT E - COMMON SPREADSHEET ERRORS 48
ATTACHMENT F – DEVELOPING TEST PLANS AND TEST CASES 50

1.0 PURPOSE AND SCOPE
(7.1.1)

This procedure provides requirements for the development, documentation, and verification of utility calculation software (specifically spreadsheets). (Spreadsheets are discussed as utility calculation software in DOE Guide 414.1-4 and elsewhere in this procedure.) This procedure applies to all Washington River Protection Solutions, LLC (WRPS) personnel (and staff augmentation) using spreadsheets for performing calculations or data manipulation in support of the preparation of technical documents or the performance of technical activities. The procedure also provides requirements for subcontractors developing spreadsheets under task-based contracts.

Spreadsheets generated by the 222-S Analytical Laboratory are subject to the requirements of this procedure plus additional Laboratory-specific requirements described in ATS-310, Section 8.14, "Computer Software Management."

Spreadsheets that are exempted from the requirements of this procedure include:

- Business or financial spreadsheets
- Spreadsheets that do not perform calculations or do not perform technical data manipulation
- Spreadsheets that are used for administrative type activities (such as table generation)
- Computation type spreadsheets as defined in Section 5.0 of this procedure
- Spreadsheets that are generated as a result of a query or download from Tank Waste Information Network System (TWINS), or other system registered on the Hanford Information Systems Inventory (HISI), provided that nothing is done to modify the data
- Spreadsheets that are generated as a result of a download or report from a software application registered on HISI into verified spreadsheet templates, provided that nothing is done to modify the data
- Spreadsheets that are generated as output from software that has been previously developed and verified in accordance with TFC-BSM-IRM_HS-C-01
- Spreadsheets that are used to check or to verify the results of another spreadsheet, software, or other calculation. When a spreadsheet is used to verify another spreadsheet, it must be developed in an independent manner, i.e., to use a different structure and different formula.

  NOTE: Although exempted from the requirements of this procedure, the spreadsheets listed above may be documented and released for tracking purposes (traceability) using a Spreadsheet Verification and Release Form (SVF) at the discretion of the manager responsible for the spreadsheet.

2.0 IMPLEMENTATION

This procedure is effective on the issue date shown in the header. All spreadsheets developed or released in documents after the issue date shall meet the requirements of this procedure. Single-use spreadsheets that were issued or released prior to the issue date will not be required to meet the documentation and verification requirements of this procedure, unless revised after the issue date. Multiple-use spreadsheets developed under previous revisions of this procedure may remain in use under their existing documentation and shall be brought into compliance with the documentation and verification requirements of this procedure at the time of their next update or revision.

3.0 RESPONSIBILITIES

- The owner of a spreadsheet shall be a technically competent individual with the appropriate education, experience, and expertise to identify the desired calculations to be performed or the desired functionality to be contained in the spreadsheet.

- The developer of a spreadsheet shall be a technically competent individual with the appropriate education, experience, and expertise to configure a spreadsheet to perform the desired calculations or to develop the requested spreadsheet functionality.

- The verifier of a spreadsheet shall be a technically competent individual with the appropriate education, experience, and expertise to determine if the spreadsheet is correctly configured to perform the desired calculations or functionality, and shall be someone who was not directly involved in the development of the specific spreadsheet revision being verified.

  NOTE: The text regarding no direct involvement in the development of the specific spreadsheet revision being verified is intended to avoid a “self-review” of a new spreadsheet or a specific spreadsheet revision by the developer. It is not intended to prevent someone who had developed a spreadsheet or a specific spreadsheet revision from verifying a subsequent revision to that same spreadsheet that was developed by someone else.

- The user of a spreadsheet shall be a technically competent individual with the appropriate education, experience, and expertise to understand the calculations, to apply the results of the calculation, or to use a spreadsheet template to perform the calculation.

  NOTE: Initial training of staff includes software quality assurance requirements and spreadsheet verification techniques. Training on software quality assurance requirements can be satisfied by completing the web-based Software Quality Assurance course (Course 356123) and training on spreadsheet verification techniques can be obtained by reviewing Attachment F of this procedure. Engineers can satisfy all the training requirements by completing the Core Engineer qualification process (Qualification Card 350850).

Other responsibilities are contained within Section 4.0 of this procedure.

4.0 PROCEDURE

4.1 Application of the Graded Approach

This procedure uses a graded approach to required software documentation and verification, depending on the software grade of the spreadsheet. The table in Attachment A of this procedure provides an overview of the software lifecycle phases, associated work activities, and documentation requirements for spreadsheets based on the grade of the spreadsheet as determined in Section 4.2. In general, spreadsheets are classed as either safety software (Grades A, B, or C) or non-safety software (Grades D, E, or N/A). Spreadsheets graded as safety software will be documented using a Spreadsheet Description Document (SDD) and issued and released with an SVF. The SVF form and Instructions and a standard template for the SDD are available from the Information Resource Management (IRM) web page (WRPS Home Page>General Information>Chief Information Officer>IRM Templates).

Spreadsheets graded as non-safety software will typically be documented only on the SVF used to issue and release the spreadsheet. Additional documentation may be provided for non-safety spreadsheets when appropriate. An example of when additional documentation may be appropriate is when a spreadsheet (or several spreadsheets) developed to support a software application are described in the Software Management Plan (SMP) for that software application.

Spreadsheet development activities are typically relatively simple compared to custom software development projects or significant configurable software projects. In applying a graded approach to documentation requirements for spreadsheets, this procedure provides standardized approaches to several aspects of spreadsheet development. This procedure, therefore, serves as the SMP, Software Configuration Management Plan (SCMP), Risk Management Plan, Contingency Plan, Installation Plan, Training Plan, and Retirement Plan and provides a standardized approach to Problem Reporting and Corrective Actions for the general use of spreadsheets by WRPS personnel.

Two reviews are required to satisfy Sections 2.3.5.6.1 and 2.3.5.6.2 of the WRPS Quality Assurance Program Description (QAPD; TFC-PLN-02) and the reviews may be combined, as appropriate to the defined software engineering method. Safety-affecting spreadsheets are developed in a manner closely approximating the waterfall methodology described in DOE G 414.1-4. Because the waterfall method is used, the first review for safety-affecting spreadsheets is performed after requirements are identified and before the utility calculation is implemented (developed) and the second review is performed after the utility calculation has been verified. Non-safety affecting spreadsheets are often developed in a more iterative approach because not all of the spreadsheet requirements can be defined at the beginning of the development process and because many of the spreadsheets are developed in a short period of time; on the order of hours to days. Because an iterative approach is used for non-safety-affecting spreadsheets, both reviews are combined and performed at the end of the spreadsheet development effort as allowed by the QAPD.

This procedure should generally be followed in the order written. However, steps may be performed in parallel or out of sequence as required.

4.2 Initial Software Project Management and Quality Planning
(7.1.1)

This procedure represents the software management and quality assurance plan for the development, implementation, and use of spreadsheets. Responsibilities for the development of spreadsheets are defined in Section 3.0 and in the individual steps of Section 4.0. Section 4.0 also describes the managerial and technical processes and work tasks required to develop, implement, and control spreadsheets.

The Software Management Plan (SMP) requirements are satisfied by the following aspects of this procedure:

- The procedure identifies all the software development and procurement tasks associated with utility calculation software.
- The procedure identifies the software work activities by phase and guides the development of utility calculations through a graded approach based on the grade of utility calculation being developed.

Responsible Manager

1. Define the problem to be solved and assign qualified personnel to fulfill the roles of spreadsheet owner, spreadsheet developer, verifier, and spreadsheet user.

   NOTE 1: Qualification requirements are defined in Section 3.0 of this procedure.

   NOTE 2: The user should start the software selection process in TFC-IRM-BSM_HS-C-01. Some of the steps within this procedure may duplicate steps in TFC-BSM-IRM_HS-C-01 for continuity within this procedure.

Spreadsheet Owner

2. Review the problem to be solved and ensure that the objectives are clearly understood and defined.

3. Review existing spreadsheets to determine if a spreadsheet has already been developed to perform the desired calculation or data analysis.

   a. If no spreadsheet exists to perform the calculation, proceed to step 4 to start the development of a new spreadsheet.

   b. If a Grade A, B, or C single-use spreadsheet exists for performing the calculation, proceed to step 8 in Section 4.3.11.1 to use the existing spreadsheet.

   c. If a Grade A, B, C multiple-use spreadsheet exists for performing the calculation, proceed to step 11 in Section 4.3.11.2 to use the existing spreadsheet.

   d. If a Grade D, E, or N/A single-use spreadsheet exists for performing the calculation, proceed to step 8 in Section 4.4.9.1 to use the existing spreadsheet.

e. If a Grade D, E, or N/A multiple-use spreadsheet exists for performing the calculation, proceed to step 10 in Section 4.4.9.2 to use the existing spreadsheet.

NOTE: Available spreadsheets can be used as long as:

- They are registered in HISI and the registration is up to date
- The planned use is within the documented purpose and scope, and the approved software grade.

If these criteria are not met, then it is considered that the available spreadsheet is being repurposed and the modification of the spreadsheet is treated the same as the development of a new spreadsheet.

Spreadsheet Owner

4. Register the new spreadsheet in HISI by creating a new HISI entry and completing the Core Information and Hanford Contacts sections within the HISI entry in accordance with the direction provided on the IRM web site.

NOTE 1: The Status field in the Core Information tab within HISI should be set to "In Development" during the initial registration of the spreadsheet. The In Development status will continue until after the spreadsheet's HISI Version Description is approved.

NOTE 2: The HISI User Guide provides guidance for completing the HISI entry including the definition of various roles within HISI and the HISI permissions associated with each role. Additional guidance for completing the HISI entry for spreadsheets may be obtained by contacting the WRPS Chief Information Officer (CIO) staff.

NOTE 3: Multiple non-safety spreadsheets developed for the same purpose that will have the same software grade may be registered under one HISI entry as long as the HISI entry identifies what spreadsheets (or types of spreadsheets) are covered by that entry. It is recommended that this information be included in the HISI Core Information tab; preferably in the Purpose/Scope field or alternately, in the Comments field.

NOTE 4: The Spreadsheet Owner may obtain an SVF number and assign a name to the spreadsheet at this point in the procedure to facilitate completion of the HISI registration rather than waiting until reaching Sections 4.3.3.1, or 4.4.1 and 4.4.3.1.

5. Complete the Software Grading Checklist (SGC) section of the HISI entry in accordance with the direction provided on the IRM web site.

   NOTE: The Spreadsheet Owner has to be assigned the role of Owner Manager or Owner Designate within the HISI entry before they can complete the SGC.

6. For new HISI registrations, contact the CIO staff assigned responsibility for Software Project Folders on the WRPS IRM Contacts web page (http://toc.rl.gov/rapidweb/cio/index.cfm?PageNum=78) to request a top-level project folder be created in Integrated Document Management System (IDMS) for the new HISI registration.

   NOTE 1: The CIO staff must create the top-level HISI Project Folder for the Spreadsheet Owner and establish the correct folder permissions.

   NOTE 2: The creation of a new HISI entry should trigger the creation by CIO staff of a HISI Project Folder in the non-record section of the IDMS. As a best management practice, in accordance with DOE Guide 200.1-1, and in response to WRPS-PER-2010-2022, use of a HISI Project Folder area has been directed for the performance of collaborative evaluations between the Software Owner and ESH&Q Software Quality Assurance.

   NOTE 3: The HISI Project Folder Area can be found through the WRPS CIO's web site; under WRPS Home, General Information, Chief Information Office, Software Quality Assurance (SQA), and IDMS (look for the text 'Project Folders in IDMS – ("Collaborative Area")' or in IDMS at http://idmsweb.rl.gov/idms/livelink.exe?func=ll&objId=150712945&objAction=browse&viewType=1#.

   NOTE 4: Details on the use of the collaborative area are left to the discretion of the spreadsheet owner. The minimum required use of the area is the creation of a folder for holding a draft SDD and a draft SVF for a spreadsheet requirements review (procedure Section 4.3.8) and for holding the verified spreadsheet, the SVF, and the SDD (including any associated approval paperwork, such as the Document Release Form) for management and QA reviews before approval (step 11 in procedure Section 4.3.10 and step 11 in procedure Section 4.4.8).

   NOTE 5: The spreadsheet owner may place additional folders in the collaborative area for managing the spreadsheet configuration during development and verification. The CIO staff can be contacted for assistance when establishing or using the Project Folder area.

HISI Approvers	7.	Review the SGC and approve the software grade, or provide comments to the Spreadsheet Owner. NOTE: The review of the SGC by the HISI reviewers is performed by the WRPS Software Review Board (SRB).
Spreadsheet Owner	8.	Respond to comments and questions from the HISI Approvers until the SGC is approved and an approved software grade is obtained.
	9.	Review TFC-ENG-DESIGN-C-10 to determine if the utility calculation is also a formal engineering calculation and if it is, comply with TFC-ENG-DESIGN-C-10 in parallel to performing the steps within this procedure.
Responsible Manager	10.	Ensure that spreadsheets are developed with the site-standard version of Microsoft Excel® available from HLAN.
Spreadsheet Owner	11.	If the approved software grade is A, B, or C (safety), proceed to Section 4.3 of this procedure.
	12.	If the approved software grade is D, E, or N/A (non-safety), proceed to Section 4.4.

4.3 Development and Use of Safety Spreadsheets

This section of the procedure describes the process to develop and use a safety spreadsheet (i.e., spreadsheets graded A, B, or C). The procedural steps are aligned with the software work activities identified in DOE G 414.1-4.

4.3.1 Software Project Management and Quality Assurance Planning

Spreadsheet Owner	1.	Obtain a copy of the SDD template from the IRM web site.
	2.	Obtain a document number for the SDD from the Hanford Document Numbering System (HDNS; http://msc.rl.gov/rapidweb/hdns).
	3.	Complete the Introduction and Purpose and Scope Definition sections of the SDD (Sections 1.0 and 2.1, respectively), based on the purpose and scope defined for the software from TFC-BSM-IRM_DC-C-01 and Section 4.2.
	4.	Complete the Software Lifecycle Activities and Grading Requirements Section of the SDD (Section 1.1).
	5.	Complete the Alternatives Analysis section of the SDD (Section 2.2) to identify if any alternatives to the use of a spreadsheet were considered for the application.
	6.	Consider the training needed for the spreadsheet and start drafting the Training Plan section of the SDD (Section 6.3).

4.3.2 Software Risk Management

The project risk inherent in the development of Grade A, B, or C utility calculation software applications is small and is generally covered by the steps within this procedure for software configuration management and verification and validation of software functionality. When additional or specific project risk management measures are considered required for a utility calculation software application, they will be included in the Project Risk Management section of the SDD.

Spreadsheet Owner

1. Complete the Project Risk Management section of the SDD (Section 2.3).

4.3.3 Software Configuration Management

This section of the procedure provides the configuration management plan for development and use of safety spreadsheets.

4.3.3.1. Configuration Item Identification.

For safety spreadsheets, the specific items under configuration management include:

- Spreadsheet File(s)
- Any add-in software used by spreadsheet
- SDD
- SVF.

These configuration items will become part of the software baseline once the software is approved in HISI and documents released into the document control system.

The following steps identify the methods to be used to identify the configuration items:

Spreadsheet Owner

1. Obtain an SVF number from HDNS (http://msc.rl.gov/rapidweb/hdns) for the utility calculation being developed.

2. Identify a file name for the utility calculation that starts with the SVF number and revision followed by a user-generated text string that ties the spreadsheet to the reason for its creation.

   NOTE 1: The user should strive to achieve a balance between the amount of information provide in the user-generated text string and total file path name constraints of 256 characters under the 32-bit version of Microsoft Windows XP. Example formats for the SVF number and revision are shown below along with one example of a user generated text string.

   - SVF-1234 Rev 0 DST Temperature History.xls
   - SVF-1234 REV 0 DST Temperature History.xls
   - SVF-1234 R0 DST Temperature History.xls
   - SVF-1234_R0_DST Temperature History.xls
   - SVF-1234-R0-DST Temperature History.xls.

NOTE 2: It is recommended that the file name include the word "draft," an alphabetical identifier (e.g., "A," "B," "C," etc.), and other text (as needed) to facilitate management of the spreadsheet configuration during the development, verification, and approval process.

NOTE 3: If, and when, this naming technique is used, the word "draft," the alphabetical identifier, and other text need to be deleted from the verified spreadsheet file name before submittal to Document Control for release.

3. Complete the Configuration Management Plan section of the SDD (Section 2.4) including the following information:

   - The spreadsheet file name
   - The SDD number, title, and revision
   - The SVF number and revision
   - Identification of any add-in software used in the utility calculation.

4. Obtain a copy of the Word template for the SVF from the IRM web site and complete Sections 1 through 7 of the SVF.

   NOTE: Only typed names and not signatures are required in Section 6 of the SVF at this point in the procedure.

4.3.3.2 Configuration Control

The Spreadsheet Developer and the Verifier are responsible to manage the configuration of the spreadsheet during the verification process. The Spreadsheet Owner may retain the responsibility for specifying the configuration management approach, in which case, the Spreadsheet Developer and Verifier are responsible for complying with the specified configuration management approach. All configuration items for a verified spreadsheet (i.e., the production version) are issued and controlled through the Document Management and Control System (DMCS) as discussed in Section 4.3.10 of this procedure. Controlled copies of spreadsheet files will be retrieved from DMCS for installation and use as discussed in Section 4.3.11.2 of this procedure. If problems or issues are discovered that require a spreadsheet to be changed, the problems are evaluated and changes are managed as discussed in Section 4.3.13 of this procedure. All changes to spreadsheets will be documented in the same manner as the original in the SDD and SVF as discussed in Sections 4.3.5, 4.3.6, 4.3.8, and 4.3.9 of this procedure. All changes to spreadsheets are verified and documented in accordance with Section 4.3.14 of this procedure prior to use. Final approvals and documentation of new versions of spreadsheets are discussed in Section 4.3.10 of this procedure.

Spreadsheet Owner

1. Continue the Configuration Management Plan section of the SDD, ensuring the following information is included:

   - A description of the default configuration management control approach as provided in this procedure.

   - If any specific configuration control methods or approaches are required, they should be documented in the Configuration Management Plan section of the SDD.

4.3.3.3 Configuration Status and Accounting

The configuration status of all spreadsheets is tracked and documented in HISI and in DMCS. Reviews of the status of documentation and spreadsheet file versions may be obtained from either system.

Spreadsheet Owner

1. Continue the Configuration Management section of the SDD ensuring the following information is included:

   - A description of the default configuration status and accounting approach as described above in this procedure.

4.3.3.4 Configuration Audits and Reviews

Periodic management reviews of safety spreadsheets shall be conducted either as part of the Management Observation Program (MOP) TFC-ESHQ-AP-C-03 or formal Management Assessment in accordance with TFC-ESHQ-AP-C-01. These reviews will consider reviews of the both the functional and physical configuration of safety spreadsheets.

Spreadsheet Owner

1. Continue the Configuration Management section of the SDD ensuring the following information is included:

   - A description of the default configuration audit and review process as discussed above.

Responsible Manager

2. Periodically conduct MOPs or Management Assessment of safety spreadsheets to ensure that they meet their functional requirements and are developed in accordance with this procedure.

4.3.4 Software Procurement and Supplier Management

Typically, there are no procurement activities associated with spreadsheet development unless a separate third-party add-in is needed.

Spreadsheet Owner

1. Complete the Software Procurement and Supplier Management section of the SDD (section 2.5).

2. If it is determined that a separate add-in needs to be purchased, exit this procedure and go to TFC-BSM-IRM_HS-C-01 to purchase the add-in, returning here after the add-in has been acquired and installed, and being sure to describe the acquisition of the add-in in the Software Procurement and Supplier Management section of the SDD.

4.3.5 Software Requirements Identification and Management

Spreadsheet Owner

1. Based on the function and scope of the problem to be solved (defined earlier in this procedure) and using the SVF template, identify the software requirements for the utility calculation by completing the Unique Spreadsheet Requirements section of the SVF template (item 6 in Section 8f) and using the information in Attachment B of this procedure as a guide.

2. Complete the Requirements Phase section of the SDD (Section 3.0) by referencing the SVF, by identifying the assumptions and limitations of the utility calculation, and by (optionally) describing the functional and software requirements in more detail than documented on the SVF.

   NOTE: When completed, the checklists in Sections 8f and 9b of the SVF constitute the requirements traceability matrix for a utility calculation.

4.3.6 Software Design

Spreadsheet Developer

1. Design the spreadsheet (software) considering the good practices described in Attachments C, D, and E of this procedure and including the following items at a minimum.

   - A “Documentation” worksheet containing the following minimum information:

     – Spreadsheet Owner Name
     – Reference to the SDD, SVF, and HISI entry.

   - A “Change Log” worksheet in multiple-use and revised single-use spreadsheets that describes the changes made to each version of the spreadsheet using the following minimum information:

     – A spreadsheet version number.
     – A description of the change(s).
     – The name of the person making the change(s).
     – The date the change(s) was (were) made.
     – The name of the person verifying the change(s).
     – The date the change(s) was (were) verified.

2. Determine whether to provide a copy of user instructions in the “Documentation” worksheet for multiple-use spreadsheets.

3. Describe the planned spreadsheet (software) design and design approach in the Spreadsheet Design Description section of the SDD (Section 4.1), including general descriptions for each worksheet and for any macros or scripts that will be used.

4.3.7 Software Safety

Per DOE Guide 414.1-4, the software safety work activity does not apply to utility calculation software types. Utility calculations are typically simple calculations where the software safety techniques and methods described in the guide could add undue burden to the development of the application and not increase the assurance that any software failure would not impact safety. However, if a safety analysis determines that the complexity of the utility calculation warrants the use of the techniques, they should be applied.

Spreadsheet Owner	1.	If the Documented Safety Analysis indicates that use of the software safety techniques and methods are warranted for the utility calculation, complete the Software Safety Plan section of the SDD by identifying the safety risks associated with a failure of the utility calculation, including the potential for obtaining an incorrect answer, and the steps that will be taken to mitigate the safety risks. NOTE: This is typically accomplished by performing a Failure Modes and Effects analysis or by a comparable technique.
	2.	If the Documented Safety Analysis (DSA) does not indicate that use of the software safety techniques and methods are warranted for the utility calculation, complete the Software Safety Plan section of the SDD (Section 4.3) by stating the software safety work activity does not apply to utility calculation software types according to DOE G 414.1-4.

4.3.8 Software Requirements Review

Spreadsheet Owner	1.	Under the top-level HISI Project Folder established earlier or in an existing HISI Project Folder, establish a working folder for the specific utility calculation being developed.
Spreadsheet Owner	2.	Submit the SDD and SVF for review by Responsible Manager and Quality Assurance personnel by placing copies of the draft spreadsheet and draft documents in the collaborative area.
Responsible Manager	3.	Review the SDD and SVF.
Quality Assurance	4.	Review the SDD to consider the requirements related to the activities of preparing the computer program for acceptance testing.
Responsible Manager/Quality Assurance	5.	If there are comments on the SDD or SVF, provide comments to the Spreadsheet Owner, otherwise, inform the Spreadsheet Owner that there are no comments.

Spreadsheet Owner

6. If there are comments, resolve reviewers' comments on the SDD and SVF, incorporate comment resolutions into the SDD and SVF, and return to step 4.3.8.2 to perform another review, otherwise proceed to Section 4.3.9.

4.3.9 Software Implementation

Spreadsheet Developer

1. Develop the spreadsheet according to the planned design described in the SDD.

   NOTE: Unit testing is performed in parallel to development of the spreadsheet on an ad hoc basis by the Spreadsheet Owner, and is not required to be documented.

2. If the spreadsheet cannot be developed according to the planned design to satisfy the stated requirements, modify the planned design so it can be developed to satisfy the requirements.

Spreadsheet Developer/ Spreadsheet Owner

3. Complete the Spreadsheet Test Plan section of the SVF (Section 9a) and the Test Case Description column in Section 9b of the SVF using the guidance provided in Attachment F of this procedure and including the following when appropriate:

   - A description of specific tests to be performed when using or revising the spreadsheet.
   - Identification of test cases that need to be run before using the spreadsheet, including the number of cases to run, instructions for running the test cases, and input parameters or User-definable settings to use when running the test cases.
   - Identification of expected results.
   - Identification of actions to take if the test cases do not provide the expected results.
   - Identify any test cases associated with optional requirements.

4. Complete the Test Plan and Test Cases sections of the SDD (Sections 5.1 and 5.2) by referencing the SVF.

   NOTE: The Spreadsheet Owner may provide a detailed description of tests in the SDD and summarize the tests in the SVF.

5. Complete the Operational Testing section of the SDD (Section 7.3) to identify any operational tests that need to be performed on multiple-use spreadsheets, what events trigger the performance of the testing (e.g., when Excel® patches are issued or whenever a spreadsheet is used) or on what frequency to perform the tests (e.g., monthly), and how to document the results of the testing.

NOTE: Operational testing of single-use spreadsheets is normally not warranted.

6. Complete the Software Training Plan section of the SDD (Section 6.3).

4.3.10 Software Verification and Validation

Verification of all new and revised spreadsheets by testing is required prior to their use or reliance on their results.

Spreadsheet Owner

1. Perform a self-check of the spreadsheet using the test plan and test cases as a guide.

2. If the self-check does not show that the spreadsheet passes the planned tests, return to Section 4.3.9 of this procedure to modify the spreadsheet to match the planned design, otherwise proceed to step 3.

3. Protect all of the worksheets within the spreadsheet without a password and submit the draft spreadsheet for verification and validation, along with the SDD and SVF.

   NOTE: Protecting all the worksheets without a password is done to prevent inadvertent changes.

4. Meet with the Verifier and walk through the draft spreadsheet and draft documentation with the Verifier.

Verifier

5. Using a hard copy printout and/or an electronic copy of the spreadsheet, verify the spreadsheet using the test plan and test cases identified in the SDD and SVF.

   NOTE 1: Verification testing must always be performed on a copy of the spreadsheet and not on the original spreadsheet in order to ensure independence of the spreadsheet developer and the verifier. Verification of the spreadsheet includes a review of the SDD and SVF for completeness.

   NOTE 2: It is recommended that the Verifier provide documentation in the SVF (in the comments column in section 9b or on a Continuation page) showing how the input data and formulae were verified.

   NOTE 3: The spreadsheet owner is responsible for maintaining the original spreadsheet and for directing its modification by the Spreadsheet Developer to address comments provided by the verifier.

6. If the spreadsheet does not pass the tests, identify issues with the spreadsheet to the Spreadsheet Owner for resolution, otherwise, proceed to step 8.

Spreadsheet Owner	7.	Return to Section 4.3.9 of this procedure to modify the spreadsheet to resolve issues identified by the Verifier.
Verifier	8.	Complete the test result portion of Section 9 of the draft SVF, the Acceptance Test Report section of the SDD (Section 5.3), and the Test Results appendix in the SDD (Appendix A), and inform the Spreadsheet Owner that the spreadsheet passed the tests, returning the SVF and the SDD to the Spreadsheet Owner.
Spreadsheet Owner	9.	Complete the remaining sections of the SDD (Sections 6.0, 7.0, 8.0, 9.0, Appendix B, if used, and any missing text) and SVF and update the discussions in existing sections.
	10.	Complete the release paperwork for the SDD in accordance with TFC-ENG-DESIGN-C-25.
	11.	Submit the spreadsheet, SDD and its release paperwork, and the SVF to Responsible Manager and Quality Assurance personnel for review by placing copies in the collaborative area established in Section 4.3.8 of this procedure.
	12.	If the spreadsheet is for the 222-S Laboratory, submit the spreadsheet and documentation for review and approval by the Laboratory Configuration Control Board (LCCB) to meet the requirements of ATS-310, Section 8.14, and document the LCCB approval of the spreadsheet in block 6 of the SVF.
Responsible Manager	13.	Review the verified spreadsheet, the SDD and its associated release paperwork, and the SVF.
Quality Assurance	14.	Review the verified spreadsheet, the SDD and its associated release paperwork, and the SVF to provide assurance of the satisfactory completion of the software development cycle, including acceptance testing.
Responsible Manager/Quality Assurance	15.	If the spreadsheet and its associated documentation is not ready for approval, provide comments to the Spreadsheet Owner, otherwise proceed to step 17.
Spreadsheet Owner	16.	Resolve reviewers' comments, incorporate the comment resolutions in the appropriate document, and return to step 11 to resubmit the revised draft for review.
Spreadsheet Owner/ Responsible Manager/Verifier/ Quality Assurance	17.	If there are no comments on the draft documents or after all comments have been resolved and incorporated, sign approval on the SDD release forms and on the SVF.

Spreadsheet Owner

18. Submit the original hard copy signed SVF (or a scanned copy of the original), an electronic copy of the spreadsheet, an electronic of the SDD, and a hard copy of the release paperwork (or scanned copy of the original) to the IRM Service Provider.

    NOTE 1: The SVF, SDD, and the SDD release paperwork are submitted for records retention and the spreadsheet is submitted for release into a managed folder.

    To transmit the electronic spreadsheet file to the IRM Service Provider either:

    a. Add a folder named with the SVF number and revision (e.g., SVF-1234 Rev 0) to the WRPS Technical Records Staging Area folder on IDMS under Waste Tank Cleanup>Tank Operations>WRPS Shared Projects, and add the electronic spreadsheet file and if scanned, the scanned copy of the original SVF to that folder. If the SVF original was not scanned, take the hard copy SVF form to the IRM Service Provider or

    b. Alternatively, create a folder labeled with the SVF number and revision on a compact disc or USB memory card, copy the spreadsheet file and if scanned, the scanned copy of the original SVF to that folder, and take the compact disk or USB memory card to the IRM Service Provider. If the original SVF was not scanned, deliver the hard copy SVF form along with the delivery of the spreadsheet file.

    NOTE 2: The SVF and associated documents will not be released until the IRM Service Provider receives the electronic copy of the spreadsheet.

    NOTE 3: It is recommended that the spreadsheet owner send an email to "^RIM DC" notifying the IRM Service Provider that the files have been posted in the WRPS Technical Records Staging Area. It is also recommended that the email message be marked as "High Importance" if there is a high priority for releasing the spreadsheet, SVF, and associated documents.

4.3.11 Software Installation and Operation

4.3.11.1 Installation and Operation of Single-Use Safety Spreadsheets

Single-use safety spreadsheets are assumed to have been developed to perform a particular calculation and are accessed from the DMCS when data are needed to support the preparation of a technical document or to support a technical activity.

Spreadsheet Owner

1. Complete the Version Description section of the HISI entry and submit for approval.

HISI Approvers	2.	Review the HISI Version Description section submitted by the Spreadsheet Owner.
	3.	If the HISI Version Description section is complete, proceed to step 6, otherwise provide comments to the Spreadsheet Owner that identify what is missing.
Spreadsheet Owner	4.	Revise the HISI Version Description to provide the missing information and resubmit the Version Description for review.
HISI Approvers	5.	Approve the HISI Version Description.
Spreadsheet Owner	6.	Complete the Post-Implementation Information section of the HISI Version Description entering the final HISI approval date for the implementation date.
	7.	Change the Status field in the Core Information tab of the HISI entry to "Operational." NOTE 1: For new spreadsheets being developed, the status will change from In Development to Operational after the initial approval of a spreadsheet. NOTE 2: For spreadsheets placed on standby in response to a Problem Evaluation Request (PER), the status will change from On Standby to Operational after approving the spreadsheet revision made to resolve the PER. NOTE 3: Section 4.3.13 provides directions on Problem Reporting and Corrective Actions.
Spreadsheet Users	8.	Open the released spreadsheet from DMCS when needed to obtain results or data that are being published in a released document.
	9.	If any errors are identified during the use of the spreadsheet, proceed to Section 4.3.13 of this procedure.
	10.	If a revision to the spreadsheet is required, got to Section 4.3.14 of this procedure.

4.3.11.2 Installation and Operation of Multiple-Use Safety Spreadsheets

Spreadsheet Owner	1.	Complete the Version Description section of the HISI entry and submit for approval.
HISI Approvers	2.	Review the HISI Version Description section submitted by the Spreadsheet Owner.
	3.	If the HISI Version Description section is complete, proceed to step 6, otherwise provide comments to the Spreadsheet Owner that identify what is missing.

Spreadsheet Owner	4.	Revise the HISI Version Description to provide the missing information and resubmit the Version Description for review.
HISI Approvers	5.	Approve the HISI Version Description.
Spreadsheet Owner	6.	Issue or install the spreadsheet in a working area by downloading the file from DMCS.
	7.	Complete the Post-Implementation Information section of the HISI Version Description by entering the HISI approval date as the implementation date.
Responsible Manager	8.	Ensure that access to multiple-use spreadsheets on shared drives is controlled such that only named users can modify the spreadsheets on the share drive.
Spreadsheet Owner	9.	Proceed to Section 4.3.12 of this procedure to train personnel.
	10.	Change the Status field in the Core Information tab of the HISI entry to Operational. NOTE 1: For new spreadsheets being developed, the status will change from “In Development” to “Operational” after the initial approval of a spreadsheet. NOTE 2: For spreadsheets placed on standby in response to a PER, the status will change from “On Standby” to “Operational” after approving the spreadsheet revision made to resolve the PER. NOTE 3: See Section 4.3.13 for directions on Problem Reporting and Corrective Actions.
Spreadsheet Users	11.	Use the spreadsheet in accordance with the training and user instructions given in the SDD. NOTE: Prior to using a multiple-use spreadsheet posted on a share area, the User needs to make sure that the spreadsheet version on the drive matches the latest verified version in DMCS.
	12.	Perform operational testing of the spreadsheet and document the testing as directed in Section 7.3 of the SDD.
	13.	If any errors are identified during the use of the spreadsheet, proceed to Section 4.3.13 of this procedure.
	14.	If a revision to this spreadsheet is required, got to Section 4.3.14 of this procedure.
Responsible Manager	15.	If the spreadsheet is no longer needed, proceed to Section 4.5 of this procedure.

4.3.12 Training Personnel on Multiple-Use Safety Spreadsheets

This section of the procedure constitutes the training plan for the use of safety spreadsheets.

Responsible Manager/Spreadsheet Owner

1. Provide the training about the spreadsheet described in the Training Plan section of the SDD (Section 6.3) to the personnel who will be using the spreadsheet.

   NOTE: Training about the spreadsheet may include a review of the spreadsheet structure, user instructions, and documentation.

Spreadsheet Users

2. Complete or receive the training before using the spreadsheet.

3. Return to step 11 in Section 4.3.11.2 of this procedure.

4.3.13 Problem Reporting and Corrective Actions for Safety Spreadsheets

Spreadsheet Users

1. Communicate the identified errors to the Responsible Manager.

Responsible Manager

2. If errors are discovered in verified quality-affecting spreadsheets that affect the data quality, take the spreadsheet out of service, notify the users that the spreadsheet has been taken out of service, and make sure the problem is documented through the PER process in accordance with TFC-ESHQ-Q_C-C-01.

   NOTE: Taking the spreadsheet out of service includes changing the Status field in the HISI Core Information tab from “Operational” to “On Standby.”

3. If errors are discovered in verified safety spreadsheets that do not affect the data quality (e.g., typographical errors), determine whether to initiate a PER or whether to track the issue internally, notify personnel about the errors, and make sure the error is resolved during the next revision of the spreadsheet.

Spreadsheet Owner

4. Determine the extent of revision required to correct the errors.

5. Go to Section 4.3.14 of this procedure to enter the process for revising the spreadsheet and the spreadsheet documentation, and for obtaining the necessary approvals.

4.3.14 Revisions to Safety Spreadsheets

Revisions to previously verified safety spreadsheets may be required for several reasons such as to update calculations, to correct errors, or to implement new criteria in multiple-use spreadsheets.

Spreadsheet Owner

1. Identify the need to revise a previously verified safety spreadsheet.

   a. If an existing safety spreadsheet is to be revised to create a spreadsheet for a different purpose such that the spreadsheet

takes on a new file name, return to step 4 in Section 4.2 of this procedure to develop a new safety spreadsheet.

b. Otherwise proceed to step 2.

2. Identify required changes or new criteria to be incorporated in the spreadsheet.

3. Obtain a copy of the previous version of the spreadsheet by downloading it from DMCS.

4. Rename the spreadsheet file name with the next highest version number.

5. Repeat the steps in Sections 4.3.1 through 4.3.8 of this procedure to update the spreadsheet documentation, requirements, and design and to obtain a review of the planned revision.

6. If the spreadsheet to be changed is a single-use spreadsheet that did not originally incorporate a change log, add a change log to the spreadsheet as a new worksheet that includes the following information:

   - Version number
   - A description of the change
   - The name of the person making the change
   - The date of the change
   - The name of the person verifying the change
   - The date of verification
   - Reference to the corresponding SVF number for both the original and the new version.

7. If the spreadsheet to be changed is a multiple-use spreadsheet, update the change log making sure to add a reference to the corresponding SVF number for the new version.

8. If the spreadsheet has been previously protected, unprotect the minimum number of worksheets necessary to incorporate the required changes.

9. Revise the spreadsheet following the applicable spreadsheet development steps in Section 4.3.9.

10. Repeat the steps in Sections 4.3.10 through 4.3.12 of this procedure to verify the spreadsheet changes, update the documentation, obtain a review of the spreadsheet and document revisions, issue the revised spreadsheet, update the HISI entry, and train personnel.

4.4 Development and Use of Non-Safety Spreadsheets

This section of the procedure describes the process to develop and use a non-safety spreadsheet (i.e., spreadsheets graded D, E, or N/A). The procedural steps are aligned with the software work activities identified in DOE G 414.1-4.

4.4.1 Software Project Management and Quality Assurance Planning

Spreadsheet Owner

1. Obtain an SVF number from HDNS (http://msc.rl.gov/rapidweb/hdns) for the utility calculation being developed.

4.4.2 Software Risk Management

The project risk inherent in the development of utility calculation software applications in Grades D, E, or N/A is small and does not require specific risk management measures beyond the direction provided in this procedure.

4.4.3 Software Configuration Management

This section of the procedure provides the configuration management plan for development and use of non-safety spreadsheets.

4.4.3.1 Configuration Item Identification

For non-safety spreadsheets, the specific items under configuration management include:

- Spreadsheet file(s)
- Any add-in software used by spreadsheet
- SVF.

These configuration items will become part of the software baseline once the software is approved in HISI and documents released into the document control system.

The following steps identify the methods to be used to identify the configuration items:

Spreadsheet Owner

1. Identify a file name for the utility calculation that starts with the SVF number and revision followed by a user-generated text string that ties the spreadsheet to the reason for its creation.

   NOTE 1: The user should strive to achieve a balance between the amount of information provided in the user-generated text string and total file path name constraints of 256 characters under the 32-bit version of Windows XP. Example formats for the SVF number and revision are shown below along with one example of a user generated text string.

   - SVF-1234 Rev 0 DST Temperature History.xls
   - SVF-1234 REV 0 DST Temperature History.xls
   - SVF-1234 R0 DST Temperature History.xls
   - SVF-1234_R0_DST Temperature History.xls
   - SVF-1234-R0-DST Temperature History.xls.

   NOTE 2: It is recommended that the file name include the word "draft," an alphabetical identifier (e.g., "A," "B," "C," etc.), and other text (as needed) to facilitate management of the spreadsheet configuration during the development, verification, and approval process.

   NOTE 3: If, and when, this naming technique is used, the word "draft," the alphabetical identifier, and other text need to be deleted from the verified spreadsheet file name before submittal to Document Control for release.

2. Obtain a copy of the Word template for the SVF from the IRM web site.

3. Complete Section 1 through 7 of a blank SVF form to generate a SVF.

   NOTE: Only typed names and not signatures are required in Section 6 of the SVF at this point in the procedure.

4.4.3.2 Configuration Control

The Spreadsheet Developer and the Verifier are responsible to manage the configuration of the spreadsheet during the verification process. The Spreadsheet Owner may retain the responsibility for specifying the configuration management approach, in which case, the Spreadsheet Developer and Verifier are responsible for complying with the specified configuration management approach. All configuration items for a verified spreadsheet (i.e., the production version) are issued and controlled through the DMCS as discussed in Section 4.4.9. Controlled copies of spreadsheet files will be retrieved from DMCS for installation and use as discussed in Section 4.4.9. If problems or issues are discovered that require a spreadsheet to be changed, the problems are evaluated and changes are managed as discussed in Section 4.4.11 of this procedure. All changes to spreadsheets will be documented in the same manner as the original in the SVF as discussed in Sections 4.4.4, 4.4.5, 4.4.6, and 4.4.7. All changes to spreadsheets are verified and

documented in accordance with Section 4.4.8 prior to use. Final approvals and documentation of new versions of spreadsheets are discussed in Section 4.4.8.

4.4.3.3 Configuration Status and Accounting

The configuration status of all non-safety spreadsheets is tracked and documented in HISI and in DMCS. Reviews of the status of documentation and spreadsheet file versions may be obtained from either system.

4.4.3.4 Configuration Audits and Reviews

Periodic management reviews of non-safety spreadsheets shall be conducted either as part of the Management Observation Program (MOP) TFC-ESHQ-AP-C-03 or formal Management Assessment in accordance with TFC-ESHQ-AP-C-01. These reviews will consider reviews of the both the functional and physical configuration of non-safety spreadsheets.

Responsible Manager

1. Periodically conduct MOPs or Management Assessment of non-safety spreadsheets to ensure that they meet their functional requirements and are developed in accordance with this procedure.

4.4.4 Software Procurement and Supplier Management

Typically, there are no procurement activities associated with spreadsheet development unless a separate third-party add-in is needed.

Spreadsheet Owner

1. If it is determined that a separate add-in needs to be purchased, exit this procedure and go to TFC-BSM-IRM_HS-C-01 to purchase the add-in, returning here after the add-in has been acquired and installed, and being sure to describe the acquisition of the add-in in Section 8e of the SVF.

4.4.5 Software Requirements Identification and Management

Spreadsheet Owner

1. Based on the function and scope of the problem to be solved and using the SVF template, identify the software requirements for the utility calculation by completing the Unique Spreadsheet Requirements section of the SVF template (item 6 in Section 8f) and using the information in Attachment B of this procedure as a guide.

4.4.6 Software Design

Spreadsheet Developer

1. Design the spreadsheet (software) considering the good practices described in Attachments C, D, and E of this procedure and including the following items at a minimum.

   - A “Documentation” worksheet containing the following minimum information:

     – Spreadsheet Owner Name
     – Reference to the SDD, SVF, and HISI entry.

- A "Change Log" worksheet in multiple-use and revised single-use spreadsheets that describes the changes made to each version of the spreadsheet using the following minimum information:

  – A spreadsheet version number.
  – A description of the change(s).
  – The name of the person making the change(s).
  – The date the change(s) was (were) made.
  – The name of the person verifying the change(s).
  – The date the change(s) was (were) verified.

2. Determine whether to provide a copy of user instructions in the "Documentation" worksheet for multiple-use spreadsheets.

3. Describe the planned spreadsheet (software) design and design approach in Section 8h of the SVF, including general descriptions for each worksheet and for any macros or scripts that will be used.

4.4.7 Software Implementation

Spreadsheet Developer

1. Develop the spreadsheet according to the planned design described in the SVF.

   NOTE: Unit testing is performed in parallel to development of the spreadsheet on an ad hoc basis by the Spreadsheet Owner, and is not required to be documented.

2. If the spreadsheet cannot be developed according to the planned design to satisfy the stated requirements, modify the planned design so it can be developed to satisfy the requirements.

3. Complete the Spreadsheet Test Plan section of the SVF (Section 9a) and the Test Case Description column for the Specific Spreadsheet Requirements section (Section 9b, Test Case ID 9b.19 and above) using the guidance provided in Attachment F of this procedure.

4. Complete the Operational Testing field in the HISI Version Description Document (VDD) tab to identify any operational tests that need to be performed on multiple-use spreadsheets, what events trigger the performance of the testing (e.g., when Excel® patches are issued or whenever a spreadsheet is used) or on what frequency to perform the tests (e.g., monthly), and how to document the results of the testing.

   NOTE: Operational testing of single-use spreadsheets is normally not warranted.

5. Consider the training needed for the spreadsheet and start drafting text for block 8m in the SVF.

4.4.8 Software Verification and Validation

Verification of all single-use and multiple-use spreadsheets by testing is required prior to their use or reliance on their results.

Spreadsheet Owner

1. Perform a self-check of the spreadsheet using the test plan and test cases using the Spreadsheet Verification and Validation section (Section 9) of the SVF.

2. If the self-check does not show that the spreadsheet passes the planned tests, return to Section 4.4.7 to modify the spreadsheet to match the planned design, otherwise proceed to step 3 within this section of the procedure.

3. Under the top-level HISI Project Folder established earlier or in an existing HISI Project Folder, establish a working folder for the specific utility calculation being developed.

   NOTE: As a best management practice, in accordance with DOE Guide 200.1-1, and in response to WRPS-PER-2010-2022, a Project Folder area has been created for the performance of collaborative evaluations between the Software Owner and ESH&Q Software Quality Assurance.

4. Protect all of the worksheets within the spreadsheet without a password and submit the draft spreadsheet for verification and validation, along with the SVF by placing copies of the draft spreadsheet and draft documents in the collaborative area.

   NOTE: Protecting all the worksheets without a password is done to prevent inadvertent changes.

5. Meet with the Verifier and walk through the draft spreadsheet and draft documentation with the Verifier.

Verifier

6. Using a hard copy printout and/or an electronic copy of the spreadsheet, verify the spreadsheet using the test plan and test cases identified in the SVF.

   NOTE 1: Verification testing must always be performed on a copy of the spreadsheet and not on the original spreadsheet in order to ensure independence of the spreadsheet developer and the verifier. Verification of the spreadsheet includes a review of any supporting SDD or SMP, if prepared, and SVF for completeness.

   NOTE 2: It is recommended that the Verifier provide documentation in the SVF (in the comments column in section 9b or on a Continuation page) showing how the input data and formulae were verified.

		NOTE 3: The spreadsheet owner is responsible for maintaining the original spreadsheet and for directing its modification by the Spreadsheet Developer to address comments provided by the verifier.
	7.	If the spreadsheet does not pass the tests, identify issues with the spreadsheet to the Spreadsheet Owner for resolution, otherwise, proceed to step 9.
Spreadsheet Owner	8.	Return to Section 4.4.7 of this procedure to modify the spreadsheet to resolve issues identified by the Verifier.
Verifier	9.	Complete the test result portion of Section 9 of the SVF (Section 9b and possibly 9c) and inform the Spreadsheet Owner that the spreadsheet passed the tests, returning the SVF to the Spreadsheet Owner.
Spreadsheet Owner	10.	Complete the remaining sections of the SVF and update the text in previously completed Sections when necessary.
	11.	Submit the spreadsheet and the SVF for review by Responsible Manager and Quality Assurance personnel by placing copies of the draft spreadsheet and draft documents in the collaborative area.
	12.	If the spreadsheet is for the 222-S Laboratory, submit the spreadsheet and documentation for review and approval by the Laboratory Configuration Control Board (LCCB) to meet the requirements of ATS-310, Section 8.14, and document the LCCB approval of the spreadsheet in block 6 of the SVF.
Responsible Manager	13.	Review the verified spreadsheet and the SVF.
Quality Assurance	14.	Review the verified spreadsheet and the SVF to consider the requirements related to the activities of preparing the computer program for acceptance testing and to provide assurance of the satisfactory completion of the software development cycle, including acceptance testing.
Responsible Manager/Quality Assurance	15.	If the spreadsheet and the SVF are not ready for approval, provide comments to the Spreadsheet Owner, otherwise proceed to step 17.
Spreadsheet Owner	16.	Resolve reviewers' comments, incorporate the comment resolutions in the SVF and return to step 11 to resubmit the revised draft for review.
Spreadsheet Owner/ Responsible Manager/Verifier/ Quality Assurance	17.	If there are no comments on the SVF or all comments have been resolved and incorporated, sign approval on the SVF.

Spreadsheet Owner

18. Submit the original hard copy signed SVF (or a scanned copy of the original) and an electronic copy of the spreadsheet to the IRM Service Provider for records retention and release.

    To transmit the electronic spreadsheet file to the IRM Service Provider either:

    a. Add a folder named with the SVF number and revision (e.g., SVF-1234, Rev 0) to the WRPS Technical Records Staging Area folder on IDMS under Waste Tank Cleanup>Tank Operations>WRPS Shared Projects, and add the electronic spreadsheet file and if scanned, the scanned copy of the original SVF to that folder. If the SVF original was not scanned, take the hard copy SVF form to the IRM Service Provider; or

    b. Alternatively, create a folder labeled with the SVF number and revision on a compact disc or USB memory card, copy the spreadsheet file and if scanned, the scanned copy of the original SVF to that folder, and take the compact disk or USB memory card to the IRM Service Provider. If the original SVF was not scanned, deliver the hard copy SVF form along with the delivery of the spreadsheet file.

    NOTE 1: The SVF will not be released until the IRM Service Provider receives the electronic copy of the spreadsheet.

    NOTE 2: It is recommended that the spreadsheet owner send an email to "^RIM DC" notifying the IRM Service Provider that the files have been posted in the WRPS Technical Records Staging Area. It is also recommended that the email message be marked as "High Importance" if there is a high priority for releasing the spreadsheet, SVF, and associated documents.

4.4.9 Software Installation and Operation

4.4.9.1 Installation and Operation of Single-Use Non-Safety Spreadsheets

Single-use non-safety spreadsheets are assumed to have been developed to perform a particular calculation and are accessed from the DMCS when data in the spreadsheet are needed to support the preparation of a technical document or to support a technical activity. Minor changes to a single-use spreadsheet can be treated as a spreadsheet revision and reuse of the single-use spreadsheet to perform an updated calculation is treated as a new calculation requiring a new HISI entry and SVF number. If a single-use spreadsheet proves valuable and is used repeatedly, the Spreadsheet Owner should create a multiple-use spreadsheet for that calculation.

Spreadsheet Owner

1. Complete the Version Description section of the HISI entry and submit for approval.

HISI Approvers

2. Review the HISI Version Description section submitted by the Spreadsheet Owner.

	3.	If the HISI Version Description section is complete, proceed to step 6, otherwise provide comments to the Spreadsheet Owner that identify what is missing.
Spreadsheet Owner	4.	Revise the HISI Version Description to provide the missing information and resubmit the Version Description for review.
HISI Approvers	5.	Approve the HISI Version Description.
Spreadsheet Owner	6.	Complete the Post-Implementation Information section of the HISI Version Description, entering the final HISI approval date for the implementation date.
	7.	Change the Status field in the Core Information tab of the HISI entry to Operational. NOTE 1: For new spreadsheets being developed, the status will change from In Development to Operational after the initial approval of a spreadsheet. NOTE 2: For spreadsheets placed on standby in response to a PER, the status will change from On Standby to Operational after approving the spreadsheet revision made to resolve the PER. NOTE 3: See Section 4.4.11 for directions on Problem Reporting and Corrective Actions.
Spreadsheet Users	8.	Open the released spreadsheet from DMCS when needed to obtain results or data.
	9.	If any errors are identified during the use of the spreadsheet, proceed to Section 4.4.11 of this procedure.
	10.	If a revision to this spreadsheet is required, got to Section 4.4.12 of this procedure.

4.4.9.2 Installation and Operation of Multiple-Use Non-Safety Spreadsheets

Spreadsheet Owner	1.	Change the Status field in the Core Information tab of the HISI entry to Operational. NOTE 1: Spreadsheets are considered to be operational after being issued through the DMCS. NOTE 2: For new spreadsheets being developed, the status will change from In Development to Operational after the initial release of a spreadsheet. NOTE 3: For spreadsheets placed on standby in response to a PER, the status will change from On Standby to Operational after issuing the spreadsheet revision made to resolve the PER.

NOTE 4: See Section 4.4.11 for directions on Problem Reporting and Corrective Actions.

2. Complete the Version Description Section of the HISI entry and submit for approval.

HISI Approvers

3. Review the HISI Version Description Section submitted by the Spreadsheet Owner.

4. If the HISI Version Description Section is complete, proceed to step 6, otherwise provide comments to the Spreadsheet Owner that identify what is missing.

Spreadsheet Owner

5. Revise the HISI Version Description to provide the missing information and resubmit the Version Description for review.

HISI Approvers

6. Approve the HISI Version Description.

Spreadsheet Owner

7. Issue or install the spreadsheet in a working area by downloading a copy from DMCS.

8. Complete the Post-Implementation Information Section of the HISI Version Description by entering the final HISI approval date as the implementation date.

9. Change the Status field in the Core Information tab of the HISI entry to Operational.

   NOTE 1: For new spreadsheets being developed, the status will change from In Development to Operational after the initial approval of a spreadsheet.

   NOTE 2: For spreadsheets placed on standby in response to a PER, the status will change from On Standby to Operational after approving the spreadsheet revision made to resolve the PER.

   NOTE 3: Section 4.4.11 provides directions on Problem Reporting and Corrective Actions.

10. Proceed to Section 4.4.10 of this procedure to train personnel.

Spreadsheet Users

11. Use the spreadsheet in accordance with training and the user instructions given in the SVF.

    NOTE: Prior to using a multiple-use spreadsheet stored on a shared drive, the User needs to make sure that the spreadsheet version on the drive matches the latest verified version in DMCS.

12. Perform operational testing of the spreadsheet and document the testing as directed in the Operational Testing field in the VDD tab in the HISI entry for the spreadsheet.

13. If any errors are identified during the use of the spreadsheet, proceed to Section 4.4.11 of this procedure.

14. If a revision to this spreadsheet is required, go to Section 4.4.12 of this procedure.

Responsible Manager

15. If the spreadsheet is no longer needed, proceed to Section 4.5 of this procedure.

4.4.10 Training Personnel on Multiple-Use Non-Safety Spreadsheets

Responsible Manager/Spreadsheet Owner

1. Provide training about the spreadsheet described in block 8l of the SVF to the personnel who will be using the spreadsheet.

   NOTE: Training about the spreadsheet may include a review of the spreadsheet structure, user instructions, and documentation.

Spreadsheet Users

2. Complete or receive the training provided before using the spreadsheet.

3. Return to step 10 in Section 4.4.9.2 of this procedure.

4.4.11 Problem Reporting and Corrective Actions for Non-Safety Spreadsheets

This section of the procedure focuses on the actions for Grade D, quality-affecting spreadsheets. If issues are identified for Grade E and N/A (non-quality-affecting) they may be handled in the same way as the Grade D spreadsheets at the discretion of the Spreadsheet Owner and Responsible Manager.

Spreadsheet Users

1. Communicate the errors observed to the Responsible Manager.

Responsible Manager

2. If errors are discovered in verified Grade D (quality-affecting) spreadsheets that affect the data quality, take the spreadsheet out of service, notify users that the spreadsheet has been taken out of service, and make sure the problem has been documented through the PER process in accordance with TFC-ESHQ-Q_C-C-01.

   NOTE: Taking the spreadsheet out of service includes changing the Status field in the HISI Core Information tab from “Operational” to “On Standby.”

3. If errors are discovered in verified Grade D spreadsheets that do not affect the data quality(e.g., typographical errors), determine whether to initiate a PER or whether to track the issue internally, notify personnel about the errors, and make sure the error is resolved during the next revision of the spreadsheet..

Spreadsheet Owner

4. Determine the extent of revision required to correct errors.

5. Go to Section 4.4.12 to enter the process for revising the spreadsheet and the spreadsheet documentation, and for obtaining the necessary approvals.

4.4.12 Revisions to Non-Safety Spreadsheets

Revisions to previously verified non-safety spreadsheets may be required for several reasons such as to update calculations, to correct errors, or to implement new criteria in multiple-use spreadsheets.

Spreadsheet Owner

1. Identify the need to revise a previously verified non-safety spreadsheet.

   a. If an existing non-safety spreadsheet is to be revised to create a spreadsheet for a different purpose so that the spreadsheet takes on a new file name, return to step 4 in Section 4.2 to develop a new non-safety spreadsheet.

   b. Otherwise proceed to step 2.

2. Identify required changes or new criteria to be incorporated in the spreadsheet.

3. Obtain a copy of the previous version of the spreadsheet by downloading it from DMCS.

4. Rename the spreadsheet file name with the next highest version number.

5. Repeat the development steps in Sections 4.4.1 through 4.4.6 of this procedure to update the spreadsheet documentation, requirements, and design.

6. If the spreadsheet to be changed is a single-use spreadsheet that did not originally incorporate a change log, add a change log to the spreadsheet as a new worksheet that includes the following information:

   - Version number
   - A description of the change
   - The name of the person making the change
   - The date of the change
   - The name of the person verifying the change
   - The date of verification

- Reference to the corresponding SVF number for both the original and the new version.

7. If the spreadsheet to be changed is a multiple-use spreadsheet, update the change log making sure to add a reference to the corresponding SVF number for the new version.

8. If the spreadsheet has been previously protected, unprotect the minimum number of worksheets necessary to incorporate the required changes.

9. Revise the spreadsheet following the applicable steps in Section 4.4.7.

10. Repeat the steps in Sections 4.4.8 through 4.4.10 to verify the spreadsheet changes, update the documentation, obtain a review of the spreadsheet and document revisions, issue the revised spreadsheet, update the HISI entry, and train personnel.

4.5 Software Retirement of Multiple-Use Spreadsheets

This section of the procedure focuses on the retirement of multiple-use spreadsheets when they are no longer needed. No specific steps are required for retirement of single use spreadsheets since they are only used to support a specific calculation and are not available for continued use.

Responsible Manager

1. Determine that a multiple-use spreadsheet is no longer needed and needs to be retired.

2. Direct personnel to retire the multiple-use spreadsheet.

3. Notify spreadsheet users that the spreadsheet is being retired.

Spreadsheet Owner

4. Remove all working copies of the spreadsheet from share areas.

5. If the spreadsheet contains any original data that cannot be found anywhere else, arrange for the retention of the original data.

   NOTE: The WRPS IRM staff can assist in determining the best way to retain original data.

6. Revise the HISI operational status of the spreadsheet from Operational to Retired.

4.6 Use of this Procedure by Subcontractors

This section of the procedure applies to subcontractors working on task-based contracts for the development and verification of spreadsheets, when the procedure is specified in the statement of work for the contract. This section of the procedure does not apply to staff augmentation subcontract engineers working under the TOC Quality Assurance (QA) program, who should follow the previous sections of the procedure as written. Outside subcontractors tasked to perform calculations or to develop spreadsheets under this procedure must meet the general requirements specified in Section 3.0. The following steps provide additional guidance to WRPS

Buyer's Technical Representatives (BTR) or technical Points of Contact (POC) and outside subcontractors on the use of this procedure.

WRPS BTR or Technical POC

1. Ensure that outside subcontractors tasked to perform calculations or develop spreadsheets under this procedure meet the requirements specified in Section 3.0.

2. Include reference to this procedure in the statement of work for contracts/tasks requiring spreadsheet development and ensure that the subcontractor personnel have access to either a hard copy of the procedure or are able to access the procedure from HLAN.

3. Ensure that subcontractor personnel have access to the HDNS (http://msc.rl.gov/rapidweb/hdns) for obtaining SVF numbers or obtain numbers for the subcontractor for each SVF requested.

   NOTE: If obtaining numbers for the subcontractor, it is preferable not to take out a block of numbers, which may then go unused, but to obtain the numbers individually at the time they are needed.

4. As requested, assist the subcontractor in submitting the SGC to the Software Review Board to obtain a software grade to ensure that the appropriate level of documentation and verification is performed.

Subcontractor Personnel

5. Follow this procedure, as written, for the development, documentation, verification, approval, and revision of spreadsheets with the following clarifications.

6. Ensure that personnel preparing and verifying spreadsheets meet the general requirements provided in Section 3.0.

7. Where concurrence is required from the spreadsheet owner's manager on the category of spreadsheet selected, obtain this concurrence from the WRPS BTR or technical POC.

8. Obtain SVF numbers from HDNS or from the WRPS BTR/Technical POC if HDNS access is not available.

9. Ensure all approval signatures on the SVF are subcontractor personnel, with the responsible subcontractor's technical manager for the task signing as spreadsheet owner's manager.

10. Where this procedure calls out TFC-ENG-DESIGN-C-10 and/or TFC-ENG-DESIGN-C-25 for documentation of calculations or technical documents, compliance with those procedures is only required if those procedures have been invoked in the statement of work for the task; otherwise, the subcontractor's applicable procedures for calculations and technical documents under their QA program should be invoked instead.

11. On completion of the spreadsheet verification, ensure that both the completed SVF and an electronic file of the spreadsheet are turned over to the point of contact specified in the statement of work.

WRPS BTR or Technical POC

12. Review and accept subcontractor's submitted calculations in accordance with TFC-ENG-DESIGN-C-10.

13. Submit subcontractor's completed SVFs, associated electronic spreadsheet files, and associated documents to the IRM Service Provider in accordance with the appropriate Section based on the software grade.

5.0 DEFINITIONS

Acceptable Excel® add-in software. There are many Excel® "add-in" software packages available that are able to provide enhancements to Excel's standard capabilities. Use of such packages without additional verification and validation is acceptable if they meet the following criteria. The "add-in" software must be commercially available software, suitable for its intended purpose, and must be a production version of the software (e.g., not be a "beta" or test version.)

Code Walkthrough. A code walkthrough is defined for this procedure as a review of the spreadsheet by the Spreadsheet Owner with the Verifier to provide an overview of the spreadsheet or the changes to the spreadsheet.

Macros. In Excel®, a macro is a module containing commands or functions: i.e., a series of actions used to automate a task. This has the following advantages: a) saving time in performing repeated actions; b) reducing the possibility of introducing errors into the spreadsheet; or c) enabling users to perform complex operations in a single step. There are no restrictions within this procedure on the use of macros written in Excel's built-in programming language VBA (Visual Basic for Applications), with the exception that the correct functioning of the macro must be verified and documented as part of the overall spreadsheet verification.

Operational Testing. Operational testing is the testing of a spreadsheet during its use to determine if it is continuing to perform correctly. Operational testing is not normally performed on a single-use spreadsheet. Operational testing of a multiple-use spreadsheet may be performed after the implementation of software patches (e.g., for Windows or Excel®), whenever a spreadsheet is used, or on a regular frequency. The need for operational testing, the tests to perform, and the timing for the performance of the testing are dependent on the particular spreadsheet.

Software Baseline. A version of a software application and its associated documentation that have been formally reviewed and agreed upon, that serves as a reference for the use of that software version or as a starting point for developing a new software version, and that can only be changed through formal change control procedures.

Spreadsheet categories. There are five categories of spreadsheets discussed in this procedure, as follows:

Computation spreadsheets. Computation spreadsheets are simple spreadsheets used for simple tasks and include only simple calculations, and must not use macros, third-party "add-in" software (i.e., not supplied by Microsoft), or contain links to other workbooks. Computation spreadsheets may be used for preliminary or rough-order-of-magnitude work, and they may not be used to support published technical documents or to support

data stored in database applications, such as TWINS. Spreadsheets used in published technical documents or in database applications or for calculations associated with safety class or safety significant equipment or to verify safety basis or environmental compliance must be defined as one of the other categories of spreadsheets below and verified in accordance with this procedure no matter how simple the calculation. The results of computation spreadsheets shall not be presented to the Office of River Protection or external regulators, unless the status of the spreadsheet results and extent of verification are clearly communicated. A recommended approach for publishing the results of a computation in a document is to include the description of the computation in the document being published and to format the presentation so the calculation can easily be verified as part of the document review and approval.

Multiple-use spreadsheets. This category of spreadsheet includes "master" spreadsheets or spreadsheet templates that are used multiple times for performing routine or standardized calculations. The category also includes spreadsheets that are accessed by multiple users. Key examples include the waste transfer compatibility spreadsheet that is used for assessing compliance of waste transfers with numerous compatibility program criteria and TSR administrative controls. Other examples include the flammable gas waste group spreadsheet and time to lower flammability spreadsheets used to perform calculations to assess compliance with TSR requirements.

Non-safety spreadsheets. Non-safety spreadsheets are spreadsheets that are graded D, E, or N/A that perform calculations that are associated with the design or operation of general service equipment or are associated with typical operational support activities or provide results that will be published in a technical document, database, or other similar software application. Non-safety also include spreadsheets used to generate graphs or tables in published technical documents. Examples include spreadsheets prepared in support of Best Basis Inventory development, general operational support spreadsheets, or spreadsheets prepared in support of individual engineering calculations for general service equipment. The category does not include safety spreadsheet applications such as spreadsheets used to perform safety basis hazard analysis calculations, calculations associated with safety-class or safety significant equipment, or spreadsheets that verify safety basis compliance.

Safety Spreadsheets. This category of spreadsheets includes spreadsheets graded A, B, or C that are used to support or perform safety analysis calculations or the design or operation of safety-class or safety significant equipment; spreadsheets that are used to classify, design or analyze nuclear facilities; spreadsheets that perform a safety function as defined in a DOE approved documented safety analysis; and spreadsheets that perform a hazard control function in support of a technical safety requirement (TSR) or protection from nuclear facility hazards.

Single-Use Spreadsheets. This category of spreadsheet includes spreadsheets used to perform a single calculation or analysis, and multiple-use spreadsheet templates populated with data from a computer model or database used to document specific modeling scenarios (e.g., the HTWOS model).

Unit Test. A unit test is the testing of an element of the spreadsheet in isolation from the rest of the spreadsheet. Examples of elements include formula, conditional formatting, lines of macro code, and custom developed functions and subroutines (macro code). In the context of this

procedure, unit testing is performed in parallel to development of the spreadsheet on an ad hoc basis by the Spreadsheet Owner, and is not required to be documented.

Utility Calculation Software. This category of software typically uses COTS spreadsheet applications as a foundation and user developed algorithms or data structures to create simple software products. The utility calculation software within the scope of this document is used frequently to perform calculations associated with the design of an SSC. Utility software that is used with high frequency may be labeled as custom software and may justify the same safety SQA work activities as custom developed software. With utility calculation software, it is important to recognize the difference between QA of the algorithms, macros, and logic that perform the calculations versus QA of the COTS software itself. Utility calculation software includes the associated data sets, configuration information, and test cases for validation and/or calibration.

6.0 RECORDS

The following records are generated during the performance of the guidance document:

- Spreadsheet Verification and Release Forms
- Spreadsheet Description Documents.
- Software Management Plans.

The record custodian identified in the Company Level Records Inventory and Disposition Schedule (RIDS) is responsible for record retention in accordance withTFC-BSM-IRM_DC-C-02.

7.0 SOURCES

7.1 Requirements

1. TFC-PLN-02, “Quality Assurance Program Description.”

7.2 References

1. DOE G 200.1-1, “Software Engineering Methodology.”
2. DOE G 414.1-4, “Safety Software Guide.”
3. TFC-BSM-IRM_DC-C-02, “Records Management.”
4. TFC-BSM-IRM_HS-C-01, “Software Development, Implementation, and Management.”
5. TFC-BSM-IRM-STD-01, “Software Life Cycle Standard.”
6. TFC-ENG-DESIGN-C-10, “Engineering Calculations.”
7. TFC-ENG-DESIGN-C-25, “Technical Document Control.”
8. TFC-ESHQ-AP-C-01, “Management and Specialty Assessments.”
9. TFC-ESHQ-AP-C-03, “Management Observation Program.”

10. TFC-ESHQ-Q_C-C-01, “Problem Evaluation Request.”

11. IEEE Standard 830, “IEEE Recommended Practice for Software Requirements Specifications,” The Institute of Electrical and Electronics Engineers, New York, New York, 1993.

ATTACHMENT A - SPREADSHEET DOCUMENTATION REQUIREMENTS

Life Cycle Phase	Work Activity	Equivalent Documentation	Safety A, B, & C[a]	Non-Safety D, E, & N/A
Concept Planning Phase	Software Project Management and Quality Planning	Scope Definition	SDD 2.1	SVF 3a
		HISI Registration/ Core Information	HISI	HISI and SVF 2e
		Alternatives Analysis	SDD 2.2	SVF 8a
		Software Grading Checklist	HISI	HISI and SVF 4
		Software Management Plan	SDD 1.0	SVF 8b
	Software Risk Management	Risk Management Plan	SDD 2.3	SVF 8c
	Software Configuration Management	Software Configuration Management Plan	SDD 2.4	SVF 8d
	Procurement and Supplier Management	Purchase Order/Statement of Work	SDD 2.5	SVF 8e
Requirement Phase	Software Requirements Identification and Management	Functional Requirements Definition	SDD 3.1	SVF 8f
		Software Requirements Specification	SDD 3.2	SVF 8f
		Requirements Traceability Matrix	SDD 3.3 and SVF 9b	SVF 9b
Design Phase	Software Design	Software Design Description	SDD 4.1	SVF 8h
		User Documentation	SDD 4.1 and 6.4	SVF 8h and 8n
	Software Safety	Software Safety Plan	SDD 4.3	NA
Implementation Phase	Software Implementation		SDD 4.0	SVF 8g
Test Phase	Software Verification and Validation	Code Walkthrough	SDD 4.4	SVF 8i and 9b
		Test Plan[b]	SDD 5.1	SVF 8i and 9a
		Test Cases[b]	SDD 5.2 and SVF 8i and 9b	SVF 8i and 9b
		Acceptance Test Report	SDD 5.3 and SVF 8j and 9b	SVF 8j and 9b
Installation Phase	Software Installation and Operation	Installation Plan	SDD 6.1	SVF 8k
		Contingency Plan	SDD 6.2	SVF 8l
	Training of Personnel in Design, Development, Use, and Evaluation of Safety Software	User Qualifications	SDD 6.3	SVF 8m
		Training Plan	SDD 6.3	SVF 8m
		User Instructions	SDD 6.4	SVF 8n
Maintenance Phase	Problem Reporting & Corrective Action	Problem Reporting & Corrective Action	SDD 7.1	SVF 8o
		Operational Testing	SDD 7.3	HISI VDD tab
Retirement Phase	Software Retirement	Retirement Plan	SDD 8.1	SVF 8p

Note: The Spreadsheet (Software) Description Document Section numbers are taken from the document template and the Section numbers may change in the issued documents.

ATTACHMENT B - DEVELOPING SPREADSHEET REQUIREMENTS

[1]Requirements describe in detail what a software system or spreadsheet is supposed to do. The collection of all requirements that define the performance of a software system or spreadsheet is known as the software requirements specification (SRS). The SRS for a utility calculation should identify all the functional requirements; the performance requirements; the security requirements, including user access controls; the interface and safety requirements; and installation considerations and design constraints where appropriate. Examples of requirements for a utility calculation include what calculation or analysis needs to be done by the spreadsheet, what data will be used as input data, does the input data need to be validated and how will it be validated, what results are to be produced, which units will be used as input or output, and what equations or formulas are to be applied to the data.

A good requirement has the following attributes:

- Correct - A requirement is correct if the performance it states meets an actual need.
- Consistent - A requirement is consistent if it does not negate, conflict with, or override another requirement.
- Unambiguous – A requirement is unambiguous if it has only one interpretation.
- Ranked for importance – A requirement is ranked for importance if it is assigned a priority level. The following priority levels are suggested.
  - Essential – the requirement must be met before accepting the software
  - Conditional – the requirement would enhance the software (spreadsheet) if met but would not make the software (Spreadsheet) unacceptable if absent
  - Optional – the requirement implies a class of functions that may or may not be worthwhile. An optional requirement gives a supplier or developer the opportunity to propose something that exceeds the customer's requirements
- Verifiable - A requirement is verifiable if it is stated in concrete terms and measurable quantities, allowing it to be objectively tested.
- Traceable – A requirement is traceable if its origin is clear and explicitly referenced in the documentation. Two types of traceability are recommended.
  - Backward traceability – A requirement is backward traceable if the source of the requirement is explicitly referenced in documents written earlier in the software development process.
  - Forward traceability – A requirement is forward traceable if the requirement can be traced to all documents spawned by the SRS.

---

[1] The content of this attachment was adapted from IEEE Std 830, 1993, IEEE Recommended Practice for Software Requirements Specifications.

ATTACHMENT B – DEVELOPING SPREADSHEET REQUIREMENTS (cont.)

When writing a requirement statement, consider how each requirement will be tested and what acceptance criteria will be used to determine that a test is successful.

A good software requirements specification (SRS) has the following attributes:

- Unambiguous – An SRS is unambiguous if, and only if, every requirement stated therein has only one interpretation.
- Complete – An SRS is considered complete when it identifies all the functionality, performance, design constraints, interface, limits, or bounds for the spreadsheet.
- Verifiable – An SRS is verifiable if every requirement stated therein can be objectively tested or observed using concrete terms and measurable quantities.
- Consistent – An SRS is consistent if the requirements do not conflict with or defeat another requirement.
- Modifiable – An SRS is considered modifiable if it can be easily changed while maintaining the characteristics of a good SRS.
- Traceable – An SRS is traceable if the requirements can be traced to their origin throughout the utility calculation software life cycle.

ATTACHMENT C - GOOD SPREADSHEET DESIGN PRACTICES

A number of good practices for spreadsheet design are well established in the literature on spreadsheet development. Following these standard practices during spreadsheet design will help ensure that the final spreadsheet is easier to understand for both a user and a verifier and significantly reduce the risk of errors in the final spreadsheet product. These practices are not mandatory but should be followed to the extent practical when developing new spreadsheets.

1. Use A Modular Design

Although at least one paper suggests building spreadsheets in which everything is contained on a single worksheet, this is not the generally accepted practice for good spreadsheet design and is not recommended. Other than for relatively small and simple spreadsheets, it is better practice to use a modular design that utilizes multiple worksheets, with each worksheet dedicated to a specific purpose. It will be much easier to manage, navigate, and verify a well designed modular spreadsheet than it is to wade through hundreds or thousands of rows and columns crammed onto a single worksheet.

- Using different worksheets, whenever different column headings are required makes it easier to expand the spreadsheet in future, avoiding the difficulty of column inserted between headings at the top of a worksheet accidentally splitting different headings located lower down the worksheet.

- If a spreadsheet contains sections which can be usefully repeated, such as a data for different waste tanks, the best way to present it in a spreadsheet is to use one worksheet for each repeatable block. By standardizing on the layout of the first worksheet, the spreadsheet can be developed quicker by copying the standardized worksheet, as required.

2. Separate Inputs, Calculations, and Results

It is good practice to provide a clear separation between the area of a workbook in which the user enters data, the area that perform calculations on that data, and the area in which the final results of the calculations are displayed. This practice makes it easier to identify all the input data used in a spreadsheet and greatly reduces the risk that calculations will be accidentally overwritten with input data. Providing distinct color coding of the data entry cells helps to clearly define the input data area of the worksheet. Providing a separate area for the calculations makes it easier to verify and maintain the spreadsheet by keeping the calculations together in a defined part of the spreadsheet. Keeping the results separate from the input and calculations makes it easier to print final reports and to clearly distinguish between results and their basis.

This separation will most typically be achieved by the use of separate worksheets: one for input data, one for calculations, and one for results. With most tank farm spreadsheets that require a significant volume of input data, this will be the preferred arrangement.

If the spreadsheet is small enough, the separation may be achieved on an individual worksheet. Two ways to achieve this are common: top-to-bottom and diagonal. In the top-to-bottom arrangement, the top of the worksheet is dedicated to the input data, the middle section of the worksheet contains the calculations, and the bottom of the worksheet displays the results. An alternate and preferred arrangement for a single worksheet is to utilize a diagonal arrangement in which each section is added below and to the right of the section above it. The advantage of this arrangement is that none of the

sections share rows or columns with each other, making it easier to add or delete rows or columns without risking impacting another section of the worksheet.

3. Keep Formulas Simple

It is good practice to avoid the creation of long complex formulas containing multiple functions or complex logic. Formulas will be much easier to understand and verify if long formulas are broken down into their intermediate steps and formulas are limited to one or two functions in a given cell. Although more rows or columns will be required to achieve this, the resulting formulas will be much easier for a verifier to follow and check, will be less prone to error, and will result in a more robust final product.

One of the most common ways that formulas become too long is by including multiple nested IF commands. Frequently, this problem can be overcome by using table lookup functions, such as VLOOKUP (vertical lookup) or HLOOKUP (horizontal lookup).

4. Use One Formula per Row or Column

As far as possible, formulas should be written so that a single formula can be copied across the entire row of calculations or down the entire column. Using one formula per row has the following advantages:

- Quicker development, because every formula can be written in a single column, then copied across all other columns
- Effective testing, especially when you use formula maps generated by software such as Spreadsheet Professional that distinguish between unique and copied formulas
- Robust updating and maintenance, because every time a formula is changed in a cell you know that it should be copied across the rest of the row. One of the most common causes of errors in formulas is when a change is made but old versions of the formula are left lurking behind
- Straightforward documentation in a specification which has one definition for each row of the final spreadsheet.

5. Avoid Creating Circular References

It is good practice to design a spreadsheet so that the logic flows from front to back, top to bottom, and left to right. This makes the spreadsheet easier to understand and reduces the risk of introducing a circular reference to the calculation.

For example, each cell reference in a formula should refer either to an input or calculation further up the same column or to an input or calculation on an earlier worksheet in the workbook, or to a calculation further down the worksheet but in an earlier column.

6. Include a Documentation Worksheet

It is good practice, and required by this procedure in Section 4.3.6 and 4.4.6, to provide some internal documentation within the spreadsheet. The extent of documentation will depend on the importance and complexity of the spreadsheet. However, the documentation will provide key information about the

ATTACHMENT C - GOOD SPREADSHEET DESIGN PRACTICES (cont.)

spreadsheet to guide future users and verifiers of the spreadsheet and avoid the need to consult the original developer of the spreadsheet during verification or use. Requirements for documentation worksheets within this procedure are discussed in Section 4.3.6 and 4.4.6.

ATTACHMENT D - SPREADSHEET HINTS AND TIPS

While not being appropriate to all spreadsheets, the ideas in this section can be useful ways to make the spreadsheet easier to understand and use.

Use Color Coding

Color coding is a useful technique to make spreadsheets easier to use and can be used to more clearly distinguish between separate sections of a spreadsheet as discussed in Attachment A. If you use dark colors for text, then black and white printouts will be largely unaffected, but the difference will be visible on screen. Using pale colors for shading cells will be visible on printouts, but may become indistinct on photocopies or faxes.

Examples of using color coding include:

- Separating inputs from calculations by making all of the cells requiring inputs the same color
- Separating types of inputs
- Highlighting cells which are linked to external spreadsheets
- Highlighting the cells modified when you made a particular change to the spreadsheet, making it much easier to track changes or, if necessary, reverse the change at a later date.

Include Units

The inclusion of a units column on every worksheet will tell you what is expected in each input cell and what is being presented in calculations. Verification of spreadsheet formulas should verify the dimensional consistency of formulas.

Use Natural Language Formulas

For simple spreadsheets, it can be useful to include an extra column containing natural language translations of the formulas in the spreadsheet.

This helps to make the calculations in a spreadsheet easy to understand, makes important logical assumptions more transparent to anyone using the model and makes it possible for an independent tester to check the coding in the spreadsheet.

When the spreadsheet development and verification is complete, the column containing the formula can be safely hidden, if desired, but not deleted to keep it as a record of what the spreadsheet is attempting to do.

Alternatively, spreadsheet auditing software packages are able to translate cell formulas into natural language formulas based on row and column headings in the spreadsheet. While potentially very powerful, caution needs to be exercised when using these translators since poor spreadsheet design and the use of subheadings can confuse the translators and make the translations incorrect. Spreadsheet Professional includes options that enable the user to specify the location of row and column headings and may be able to overcome this limitation.

ATTACHMENT D - SPREADSHEET HINTS AND TIPS (cont.)

Use Named Ranges

Consider using range names when developing a spreadsheet.

Allocating meaningful range names to areas or cells within a spreadsheet can make the formulas in a spreadsheet easier to understand and reduce the risk of errors made by referring to the wrong cell. This is most useful when a cell is referenced in a formula a long way from where it is calculated, particularly when the input cell is contained on a different worksheet than the formula.

When referring to nearby cells, it is preferable to keep with cell references rather than range names. Having the referenced cells close to their point of reference makes it relatively easy to trace and understand the formula, and avoiding range names will keep the formula definition shorter.

If using range names, include a table of range names and their associated cells on the documentation worksheet of the spreadsheet.

Build in Error Traps

It is good practice to include checks for errors in a spreadsheet. These can be set up to check for errors in the input data provided, or for errors in the spreadsheet formulas.

Cross casting is the process of checking that the totals in the rows and columns of a spreadsheet are consistent. This can prove useful for detecting errors introduced by the insertion of rows or columns.

Use Data Validation

Data validation is a process of ensuring that a spreadsheet operates on clean, correct and useful input data. Data validation testing looks for too much data, too little data, no data, the wrong kind of data, the wrong size of data, and inconsistencies in the data and notifies the User of an error or closes the spreadsheet process without crashing or without allowing unauthorized access to the User's computer.

- The data validation feature of Excel®, available under Data>Validation, allows the user to specify valid data ranges and types and the message to display if inputs lie outside the specified range or data type.

- Conditional formatting is an alternate method for data validation and is used to identify when a result is outside an acceptable range by changing the color of the cell.

- The use of error trapping code in macros or scripts can be used for data validation by looking for exceptions resulting from out-of-range data values and handling the exception in a controlled manner, preventing the potential for exploitation of vulnerabilities in unhandled exceptions.

ATTACHMENT D - SPREADSHEET HINTS AND TIPS (cont.)

Macros

- Macros can be used to automate repetitive tasks within a spreadsheet and can improve the spreadsheet usability. However, care needs to be taken to ensure that the correct functioning of the macro is verified. A useful paper may be found at http://www.css.qmul.ac.uk/training/manuals/ExcelVBAIntro.pdf.

ATTACHMENT E - COMMON SPREADSHEET ERRORS

The following represent some of the most common spreadsheet errors. It pays to be alert for these common errors during verification of the spreadsheet.

Formulas Not Copied

One of the easiest ways to introduce an error into a spreadsheet is to update the formula in a cell and forget to copy the new formula across into the other cells in a row. By using a formula map and a design in which all formulas are copied across all rows, this mistake can be noticed very quickly.

Referencing the Wrong Cell

Nearly every formula in a spreadsheet refers back to another input or calculation. With the quantity of references in any large spreadsheet, it is inevitable that you will make mistakes and refer to the wrong cell. Sometimes, the resulting formula will produce a meaningless result making it easy to spot with some simple numerical testing. If you are unlucky, the error in the result will be more subtle.

The only way to check for wrong references with any confidence is to check every unique formula in the spreadsheet. In Excel®, checking references is made easier if you use the auditing toolbar. This allows you to trace the precedent cells graphically.

Summing Over the Wrong Range

A similar mistake is to include the wrong cell reference in a SUM formula. It is particularly easy to introduce this error when you insert an extra row in a block of cells that are being summed. Insert a row in the middle of the block and the formula will automatically adjust to include the extra row, but insert a row immediately above or below the block, and the new row will be omitted from the formula.

You can build internal checks of the sums in a spreadsheet by using cross casting, as described in Attachment B. When testing a spreadsheet, Excel®'s trace precedents feature helps to find errors of this type by showing them graphically.

Confusion Between Relative and Absolute References

Another commonly found error is caused by confusion between relative and absolute references. A cell reference in a formula of the form '=D4' will change if you copy the formula across the row to E4, F4 and so on. Copied down a column it will change to D5, D6, etc. If you use the reference 'D4' it will not change when copied across or down. You can also use semi-absolute references of the form $D4 or D$4.

The most common mistake is using a relative reference instead of an absolute one. This is quite easy to spot numerically, but if you use an absolute reference in place of a relative one it can be much more difficult to spot. Again, the only way to find this mistake reliably is to go through all of the unique formulas in the spreadsheet.

ATTACHMENT E – COMMON SPREADSHEET ERRORS (cont.)

Units Errors

Mixing up the appropriate units for the elements in a calculation is another frequently occurring problem. Finding such errors can be helped by features such as inclusion of a units column to make tracing through the calculation clearer. It is also good practice to try to avoid switching between units, except when absolutely necessary.

Performing a dimensional analysis of formulas will ensure that errors in units are identified.

Misusing Functions

Certain functions such as the lookup and reference functions, VLOOKUP, HLOOKUP, INDEX, and MATCH are frequently used incorrectly. It is important to construct the VLOOKUP statement correctly to ensure that an exact rather than an approximate match to the desired value is obtained.

Complicated IF statements, especially nested IF statements, are also particularly prone to error. These functions are often very useful, but make sure that you understand exactly how they work before using them in your spreadsheet. By making the formulas in your spreadsheet easy to understand, you reduce the risk of introducing errors and increase the chances that a verifier will find your mistakes.

Spreadsheet auditing software is able to check for many function problems.

Corrupted Links to External Workbooks

Beware of making structural changes, such as inserting and deleting rows/columns, to linked workbooks. It is essential to have both workbooks open when making structural changes to a called workbook to ensure that the change is reflected in the workbook containing the link.

Hidden Cells/Pages

Hidden cells, pages, or ranges can cause difficulty both during the construction of a spreadsheet and during the verification. If you are developing or modifying a spreadsheet, hidden cells may result in the inclusion of more than is expected in a calculation, such as a SUM. Likewise, if you copy across hidden areas, you may well overwrite formulas that are important to the spreadsheet.

Additional Information

Additional useful information discussing how errors happen in Excel® can be found using the following link: http://www.auditexcel.co.za/errors.html

ATTACHMENT F – DEVELOPING TEST PLANS AND TEST CASES

A spreadsheet test plan is developed by identifying all the tests needed to demonstrate (verify) that the stated spreadsheet requirements have been met or satisfied. A spreadsheet test case is developed by identifying the conditions or variables that will be used when performing a test. Test cases may be used when testing a programming decision logic or may involve introducing wrong data types or out of range data values to test error trapping and conditional formatting.

Verification of all spreadsheets by testing is required prior to their use or reliance on their results. Verification testing consists of three stages: (1) reviewing and verifying the assumptions/limitations of the spreadsheet are reasonable and adequate; (2) verifying that the cell formulas included in the spreadsheet accurately reflect and perform the intended functions, equations, and calculations; and (3) verifying that input data used in the calculations were correctly entered into the spreadsheet. The following represent good practices for developing spreadsheet test plans and test cases.

General Test Plan Considerations

1. The error checking and formula auditing tools in Excel® should be used to detect any obvious errors in spreadsheet formulas.

2. All formulas contained in the spreadsheet should be examined to make sure they correctly implement the intended functions or calculations using the guidance provided in Appendix E and provided below to avoid commonly found errors.

3. Examine all the input data entered into the spreadsheet using appropriate techniques (see the Input Data section below).

4. "Master" multiple-use spreadsheets will not normally have input data that requires verification. However, there may be situations in which a "master" spreadsheet template contains initial or permanent data values need to be checked.

Formula Testing

1. For each unique cell formula, review the cell formula to determine that it performs the intended algebraic equation or logical expression and for non-unique formulae (i.e., the same formula applied to a series of data values), review the overall series for consistency to ensure the formulae have not been corrupted or incorrectly copied. This method would not be sufficient as the only testing method used for multi-use spreadsheets and must be supplemented with the appropriate method below.

2. Perform an alternate calculation by hand, alternate spreadsheet, Mathcad, or other means, and compare the results of the alternate calculation to the spreadsheet results to verify that the spreadsheet performs the intended functions and returns the correct results.

3. If previously published and verified data is available, use this data as test data for the spreadsheet by entering the published input data into the spreadsheet and confirming that the spreadsheet returns the same result as published.

ATTACHMENT F – DEVELOPING TEST PLANS AND TEST CASES (cont.)

4. Compare spreadsheet results with field measurements or analytical data, when appropriate, to confirm that the spreadsheet results are in reasonable agreement with the measured values.

NOTE: Alternate calculations (2) and testing (3) are the preferred methods to verify formula. When performing hand calculations or other alternate calculations to verify a spreadsheet, the results of the hand calculation should be "close enough" to the spreadsheet results to give confidence in the reasonableness of the results. This qualification is added since discrepancies in rounding or methodology may result in small differences, up to 5%. Care needs to be exercised when such differences are observed to make sure the difference is due to rounding or methodology rather than an error in the spreadsheet.

Input Data Testing

1. Identify the degree of data entry validation that will be performed.

2. For non-safety spreadsheets, check manual entries for accuracy by checking, at a minimum, a sampling of the input data entries against the original source of the information as indicated below.

    a. For spreadsheet files with less than 100 data elements, verify all input items and correct all errors found.

    b. For spreadsheet files with 101-500 data elements, verify that a 25% sample of input data taken at random is correct. If the observed error rate is less than 1% of the sample, complete the verification by correcting the errors. If the error rate exceeds 1%, correct the errors and draw another 25% sample and repeat the procedure. If the error rate for the second sample is less than 1%, complete the verification by correcting the errors. If the error rate for the second sample exceeds 1%, correct the errors and verify that all the remaining data elements are correct.

    c. For spreadsheet files with over 500 data elements, verify that a 10% sample of the input data taken at random is correct. If the error rate is less than 1% of the sample, complete the verification by correcting the errors. If the error rate exceeds 1%, correct the errors and draw another 10% sample, and repeat the procedure. If the error rate for the second sample is less than 1%, complete the verification by correcting the errors. If the error rate for the second sample exceeds 1%, correct the errors and verify that all the remaining data elements are correct.

3. For safety spreadsheets, check 100% of manual data entries for accuracy by comparing the entered data to the original source of the information.

4. For all spreadsheets, test data sets copied and pasted from another spreadsheet or application (e.g., data downloaded from TWINS), by checking the first and last values for appropriate placement and alignment in the spreadsheet cells and checking that the sum of copied cells matches the sum of the original cells.

5. For all spreadsheets, test data transferred by an automatic process by an appropriate method from above depending on the amount of data transferred.

ATTACHMENT F – DEVELOPING TEST PLANS AND TEST CASES (cont.)

6. Equivalent tests for large blocks of data include using conditional formatting, formula to compare two data values, and copying one set of data and pasting on a second set of data using Paste Special and selecting Difference.

Macro Testing

Testing of the macros should demonstrate the following:

1. The macro correctly performs the intended function.

2. The buttons, dialog boxes, and menus work as intended.

3. All the cell references within the macro are correct.

4. Macros that manage files correctly handle duplicate or non-existent files

5. The User is provided with an opportunity to change default macro settings, such as worksheet names.

6. The macro correctly handles incompatible data types and includes error-trapping to avoid damaging crashes.

7. The macro closes appropriately when the User presses Escape during macro execution.

NOTE: Verification of the macro does not necessarily require checking the line by line VBA code. Verification may be performed by checking that the macro performs its intended function either by testing it or by using an alternate means to verify its correct performance.