SOFTWARE CONTROL PROCEDURE

ATTACHMENT

G

TITLE: SCP-312, DETERMINATION OF SOFTWARE REQUIREMENTS

1.0 OBJECTIVE

The objective of this software control procedure is to provide a systematic method for determining software class, design input, design documentation requirements, testing options, and other applicable software control procedures.

2.0 APPLICABILITY

[This software control procedure is a revision and compilation of the procedures previously denoted as SCP-302, Software Development Specification Preparation and Approval; and SCP-312, Determination of Software Classification.]

This is the initial procedure for addressing software quality assurance, and is used for specifying requirements for

o software design, development, documentation, review, control, testing and use

o control of data used as input to software

o transfer of software, data and/or documentation to and from the research project.

Details of applicability shall be determined by research project planning documents, sponsor requirements, and intended end-use of software.

This procedure is applicable to all software, with the exception of software used as part of measuring and test equipment (M&TE), software encompassed by a technical procedure that prescribes methods for data acquisition, word processing software, and operating system software.

3.0 DEFINITIONS

This section contains definitions that are common to two or more software control procedures.

3.1 Acquired software and/or design documentation - Software and/or design documentation obtained by procurement or transfer from outside the research project.

QAO Approval		Approved (Research Director)	
G. Hughes 4/3/86		_V. Blowlock_ 4/4/86	
Prepared by		Technical Review	
J. Murnaghey 4/3/86		_S. Harris_ 4-3-86	
Procedure No. SCP-312	Revision No. 1	Date Issued JUN 1986	Page 1 of 15

3.2   Application - See application run.

3.3   Application run - Use of software to perform calculations or to manipulate data. Same as application.

3.4   Backup copy - A copy of a data file, software, etc. on magnetic media or as a computer listing that is retained in the event that the original copy is destroyed or lost.

3.5   Benchmarking - A type of verification in which a test problem (including input and output results) is used to assure correct model operation or to compare software.

3.6   Class determination - Designation of software into a category. The selection of a category in turn determines other requirements.

3.7   Code - See software.

3.8   Code custodian - A person designated to be responsible for accomplishing the actions required for configuration management; this individual is generally the main point of contact and authority for a given computer code.

3.9   Computer model - Engineering/scientific software and data.

3.10   Configuration management - 1. A system for orderly control of software, including methods used for labeling, changing, and storing software and its associated documentation. 2. The systematic evaluation, coordination, approval or disapproval, and implementation of all approved changes in an item of software after establishment of its configuration management (taken from DOD-STD-480A).

3.11   Conversion testing - Testing performed to assure that calculated results obtained with software installed on a specific computer are consistent with results obtained on the computer on which the software was originally developed and tested.

3.12   Data - 1. Representation of facts/concepts in a formalized manner suitable for communication, interpretation or processing by human or automatic means. 2. Any representation such as characters or analog quantities to which meaning is or might be reassigned. 3. Same as input data, numeric data, output data.

3.13   Data base - A logically unified collection of information stored on magnetic media and maintained by a research project. Within the SCP series, a flat (sequential) file or a binary worksheet (e.g., from SAS or MINITAB) containing a collection of information in a fixed configuration is considered a data base when it is used as the information base for a research project.

3.14 Data base software - Software that handles storage and retrieval of information in a data base. This software is often known as a data base management system (DBMS). Analytical software (e.g., SAS or MINITAB) is regarded as data base software when it is used to manage a data base.

3.15 Data base steward - A person designated to be responsible for accomplishing the actions required for configuration management of a data base.

3.16 Deficiency - Failure to develop, document or implement effectively any applicable element of the QA program or project activity established by mutual agreement with the sponsor, or failure to follow established procedures.

3.17 Design documentation - 1. For engineering/scientific software, documentation of software design that includes a description of mathematical models and numerical methods, and a user's manual. 2. For support software, documentation of software that includes at a minimum a user's manual.

3.18 Design input - 1. Input to the software development process, including bases for software design, functional requirements, performance requirements, regulatory requirements, and codes and standards (taken from ANSI/ASME NQA-1-1983, Supplement 3S-1). 2. Also termed "software requirement specifications": functions, performances, design constraints and attributes of software and external interfaces (taken from IEEE std 730-1984).

3.19 Documentation - Design documentation and exhibits, memos and/or other information used to assure traceability and reproducibility of software development, review, control, testing and use.

3.20 Engineering/scientific software - Software that reads input data, computes results and provides output calculations for use in performing an analysis or making an inference. Engineering/scientific software may be transferred from outside PNL or outside the research project, or it may be developed at PNL. Transferred software may be used as acquired, or it may be modified at PNL. (NOTE: Does not include system maintained software, such as LOTUS 1-2-3, RS/1, SAS, or DISSPLA, or command files written to utilize such software.)

3.21 Final Internal Development Review (FIDR) - A formal review process that compares modified or developed engineering/scientific software and its documentation to its design input and design documentation requirements, evaluates the technical validity of the software, and approves software for configuration control and verification and/or validation.

3.22 Hard copy - A computer-produced copy of information in human-readable form on paper or microfiche (as opposed to a copy on magnetic media in computer-readable form).

3.23 Incident - Any deviation from the planned or expected behavior of an activity or operation; or a course of events which has or may have a significant programmatic, safety, health, or environmental impact. Significant programmatic impacts include those associated with reliability, cost, schedule, data loss, or questions of data validity or analysis.

3.24 Independent technical review (ITR) - A documented critical review by qualified independent personnel to provide assurance that information is correct and satisfactory.

3.25 Internal testing - Informal testing of software that is performed during the development process. Internal testing does not replace verification but may be used to support the verification process.

3.26 Magnetic media - Tapes, discs or diskettes used to record and store information in computer-readable form.

3.27 Operating system software - A collection of software remaining permanently on a computer to provide overall coordination and control of the operation of the hardware. This collection includes compilers, link editors and similar software.

3.28 PAP - Acronym for PNL Administrative Procedure.

3.29 Project manager - A person designated as the manager of a research project. The term project manager also refers to those persons designated by the project manager to act on his/her behalf for specific activities.

3.30 Production software - Software for which the detailed design can be prespecified to a level of detail acceptable for development (as opposed to research software).

3.31 Program - See software.

3.32 QAD - Acronym for Quality Assurance Department.

3.33 Research project planning documents - Documents that specify agreements between PNL and a sponsor regarding the nature of work to be performed in a research project. Examples of these documents are the Project Management Plan (PMP), the Technical Program Plan (TPP), the Quality Assurance Plan (QA Plan), the Statement of Work (SOW), and the Field Task Proposal/Agreement (FTP/A).

3.34 Research software - Software for which the detailed design is being researched in the software development process, and for which comprehensive, accurate prespecification of design detail usually is not possible (as opposed to production software).

3.35 SCP - Acronym for software control procedure.

3.36 Secure storage - Controlled access, limited to individuals that are authorized for specific purposes.

3.37 Software - A sequence of instructions suitable for processing by a computer. Same as program, code.

3.38 Software development - The process by which new software (or a software segment) is created, including modification of the logic of existing software.

3.39 Stream of commands - A sequence of instructions executing system maintained software that is supplied by the user for an application run.

3.40 System maintained software - Software that is installed and maintained at the computer system level rather than at the user level, but that is peripheral to the operation of the hardware (e.g., commercial software such as LOTUS 1-2-3, RS/1, SAS, or DISSPLA).

3.41 Support software - 1. Software that may be easily and exactly verified, and that performs a simple function such as conversion of units, change in data format or plotting of data in support of engineering/scientific or system maintained software. 2. A stream of commands or sequence of streams of commands executed to utilize system maintained software, in which the system maintained software generates reportable results.

3.42 User's manual - Documentation of software that supplies information to the user to allow preparation of input and understanding of format and/or content of output.

3.43 Validation - 1. A demonstration that a computer model (data and software) adequately describes physical reality over the range of variables of interest. 2. Assurance that a model as embodied in a computer code is a correct representation of the process or system for which it is intended (taken from NUREG-0856).

3.44 Verification - 1. A demonstration that software correctly solves mathematical equations and performs the data processing it was designed to perform. 2. Assurance that a computer code correctly performs operations specified in a numerical model (taken from NUREG-0856).

3.45 Version - An item of software or documentation that is identifiably different from the original item.

4.0 RESPONSIBILITIES

4.1 Project Manager

o Assure that the appropriate Software Requirements Form (Exhibit 1, 2 or 3) is completed.

o Assure that an independent technical review (ITR) of the Software Requirements Form (Exhibit 1, 2 or 3) is performed.

o Obtain sponsor approval of tne Software Requirements Form, Engineering/Scientific Software (Exhibit 1), if required by research project planning documents.

o Assure that copies of all research project records generated as a result of implementing this SCP are processed appropriately.

4.2 Preparer of the Software Requirements Form

o Fill out the appropriate Software Requirements Form (Exhibit 1, 2 or 3).

o Approve the Software Requirements Form (Exhibit 1, 2 or 3) by signature and date.

4.3 QAD Representative

o Review and approve the Software Requirements Form, Engineering/ Scientific Software, Exhibit 1.

5.0 PROCEDURE

5.1 Introduction

5.1.1 Relationship of SCPs to NQA-1

This procedure provides the framework for compliance of the SCPs with ANSI/ASME NQA-1-1983, Quality Assurance Program Requirements for Nuclear Power Plants, Supplement 3S-1, Supplementary Requirements for Design Control. PNL is differentiating research software from production software, in that design input for production software is usually well defined. In the case of research software, the design process itself may be the primary objective of the research. Formal control of software need occur only after the design has been determined and the expected application has been specified. Freedom to try different algorithms and code pathways must be permitted during the formative stages of design. PNL has recognized this stage as research software design, where such freedom is permitted. This stage terminates with FIDR at which time the software and design documentation are baselined and placed under configuration management.

If the software is production software rather than research software, the SCPs have been designed as minimum requirements and provide the flexibility to require additional standards and specifications (e.g., IEEE-std-730-1984). In addition, the SCPs

are designed to allow additions or exceptions to specific requirements either to accommodate equivalent means of maintaining traceability and reproducibility, or to tailor the stringency of requirements to the needs of a particular piece of software.

NQA-1 specifies that for hardware design control, the following shall be addressed:

- o *design input*
- o *design process, including design analysis*
- o *design verification*
- o *change control*
- o *interface control*
- o *documentation and records.*

These SCPs are an interpretation of these hardware requirements for software.

Design input is specified in research project planning documents and on the Software Requirements Form in Sections II, IV, V and VI of Exhibit 1 and Sections II through IV of Exhibit 2. *Design process* consists of development and internal testing of software, development of benchmark test cases, and documentation of software in accordance with design input. The design process complies with the requirements specified in research project planning documents and in the Software Requirements Form (Exhibit 1), and is finalized by the FIDR. The design input (requirements for the design process) receives an independent technical review (ITR). *Design analysis* is accomplished through an in-depth technical review in accordance with SCP-313, Final Internal Development Review of Software and Documentation. When the software and its design documentation have been approved by FIDR, configuration management of and changes to software and design documentation (including *change control*) are controlled by SCP-314, Software Configuration Management. The final phase prior to releasing software for use is *design verification*, which is accomplished by following SCP-315, Software Conversion Testing, Verification and/or Validation. Software-related aspects of *interface control* are handled by PAP-101, Communication and Commitment (Interface) Control, and by specific sections of SCPs. *Documentation* generated and required to be a record is specified in each SCP. *Records* are processed according to records control procedures.

Additionally, SCPs are provided to document and review application runs (SCP-316, Software Application Control), to transfer software or data to and from the research project (SCP-317, Transfer of Software, Data and/or Documentation), and to handle data base management (SCP-318, Control of Data Bases). SCP-318, Control of Data Bases, does not manage the data during its generation; ensuring quality and integrity of data during experimentation is controlled by a project-specific technical procedure. Rather,

this procedure is meant to assure the integrity of the data during its analysis and/or use in modeling.

5.1.2 Reporting of Preliminary Results

The process of taking software from design input to design verification may be lengthy; thus, for purposes of providing research project progress to a sponsor, preliminary results may need to be reported. The project manager shall assure that any results reported to the sponsor using engineering/scientific software that has not been reviewed in accordance with SCP-313 (Final Internal Development Review of Software and Documentation) and verified in accordance with SCP-315 (Conversion Testing, Verification and/or Validation of Software) are clearly marked with the following:

Results are based on the use of unverified software. No assurance is expressed or implied as to the accuracy, completeness or usefulness of this information.

5.2 Preparation of the Software Requirements Form

5.2.1 For all software [except software used as part of measuring and test equipment (M&TE), software encompassed by a technical procedure that prescribes methods of data acquisition, word processing software and operating system software], the preparer of the Software Requirements Form shall designate one of the three classes of software by selection of the appropriate Software Requirements Form:

o Engineering/Scientific Software, Exhibit 1

o Support Software, Exhibit 2

o System Maintained Software, Exhibit 3.

Engineering/scientific or system maintained software can be used to generate reportable results. Support software must be used with engineering/scientific software, system maintained software, or another source of original data to produce reportable results.

Each application run of system maintained software used to generate reportable results requires creation and documentation of a stream of commands (e.g., command files to submit batch runs) tailored to the application. A stream of commands is written in the language of the system maintained software or in a programming language. All such streams of commands and interactions with software must be classed as support software.

5.2.2   The project manager shall assure that a Software Requirements Form
is completed for all research project software in the format of
Exhibit 1, 2 or 3. Directions for completing each form are in
italics on the form. The possible resulting sequences for
applying the SCPs for software are indicated in the flowcharts
given in Exhibits 1, 2, and 3. Exhibit 4 indicates the sequence
of applying the SCPs for data used as input to software.

5.2.3   The preparer of the Software Requirements Form (Exhibit 1, 2 or 3)
shall indicate the name of the code custodian appointed in
accordance with PAP-205, Quality Assurance Plans, in item 2 of the
appropriate Software Requirements Form.

5.2.4   The preparer of the Software Requirements Form (Exhibit 1, 2 or 3)
shall assure that at a minimum the appropriate software class,
design input, design documentation requirements, testing options,
and applicable SCPs are identified.

5.2.5   The preparer shall decide if any additions or exceptions to
requirements (entire SCPs or parts of SCPs) are needed. If so,
additions and exceptions and their explanations shall be recorded
in one of the following:

o   Section VII of Exhibit 1, for engineering/scientific software

o   Section V of Exhibit 2, for support software

o   Section IV of Exhibit 3, for system maintained software.

5.2.6   If software is proprietary, requirements that violate the software
license shall not apply. Such exceptions to requirements shall be
noted and explained in the additions and exceptions section of the
appropriate Software Requirements Form (see Section 5.2.5).

5.3   Approval of the Software Requirements Form

5.3.1   The preparer shall review designation of software class, design
input, design documentation requirements, testing options, and
other applicable SCPs; shall assure completeness of the Software
Requirements Form (Exhibit 1, 2 or 3); and shall indicate approval
by signature and date. The preparer shall forward the approved
form to the project manager.

5.3.2   The project manager shall assure that an ITR of the Software
Requirements Form (Exhibit 1, 2 or 3) is performed in accordance
with PAP-604, Independent Technical Review. The ITR shall
evaluate at a minimum the following:

o   appropriateness of software class

o   design input

o   design documentation requirements

o   testing options

o   applicable SCPs

o   completeness of the form.

The project manager shall assure that the reviewer(s) signs and dates the approval section of the Software Requirements Form. NOTE: If the project manager is not the preparer of the Software Requirements Form (Exhibit 1, 2 or 3), the project manager may perform the ITR, insofar as is practical.

5.3.3   The Software Requirements Form, Engineering/Scientific Software (Exhibit 1), shall be approved (by signature and date) by a QAD representative to assure that the form has been prepared, reviewed and approved in accordance with documented procedures and QA requirements.

5.3.4   For engineering/scientific software, if required by research project planning documents, the project manager shall forward a copy of the internally approved Software Requirements Form, Engineering/Scientific Software (Exhibit 1), to the sponsor for final approval in accordance with PAP-101, Communication and Commitment (Interface) Control. The sponsor shall be requested in a cover letter to do one of the following within a specified time period:

o   if recommendations are acceptable, approve the Software Requirements Form, Engineering/Scientific Software (Exhibit 1), by signature and date, and return the form to the appropriate PNL manager.

o   if recommendations are not acceptable, describe objections in writing and return them with the form to the appropriate PNL manager.

5.3.5   For support software and system maintained software, sponsor approval on the Software Requirements Form (Exhibit 2 or 3) is not required.

5.3.6   If the sponsor does not concur with the original Software Requirements Form, Engineering/Scientific Software (Exhibit 1), the preparer shall consider sponsor recommendations and revise the Software Requirements Form in accordance with Section 5.2, and shall have the form approved in accordance with Sections 5.3.1, 5.3.2, 5.3.3 and 5.3.4.

5.3.7 Upon final approval of the Software Requirements Form (Exhibit 1, 2 or 3), the project manager shall assure that a copy of the form is processed as a research project record.

5.4 <u>Identification of Design Input Requirements for Engineering/Scientific Software</u>

Section 5.4 applies only if development or modification of engineering/ scientific software and/or design documentation is required (see questions 8 and 9 on the Software Requirements Form, Engineering/Scientific Software, Exhibit 1).

5.4.1 The preparer of the Software Requirements Form, Engineering/ Scientific Software (Exhibit 1), shall reference any design input specified in the research project planning documents by indicating document name and page in question 18, Section IV of the form.

5.4.2 If the following design input requirements are not addressed in research project planning documents, the preparer shall evaluate the need for their inclusion:

o bases for design (e.g., physical and chemical phenomena to be accounted for or known to be neglected; input or output formats)

o performance requirements (e.g., maximum CPU time, memory requirements)

o regulatory requirements (e.g., NUREG-0856 or other specified requirements)

o codes and standards (e.g., IEEE-std-730-1984).

Requirements for additional design input, including internal testing and benchmark test cases, shall be based primarily upon the intended end-use of software, its relative importance to research project results, and sponsor requirements. Exclusions shall be documented in question 19, Section IV of the Software Requirements Form (Exhibit 1).

5.4.3 If additional design input is determined to be required, the preparer of the Software Requirements Form (Exhibit 1) shall attach the additional input and indicate in question 19, Section IV, that additional design input has been appended.

5.5 <u>Identification of Design Documentation Requirements</u>

Design documentation requirements (i.e., user's manual and mathematical models and numerical methods) shall be based primarily upon intended end-use of software, its relative importance to research project results, and sponsor requirements.

5.5.1 The preparer of the Software Requirements Form (Exhibit 1 or 2) shall determine design documentation requirements in Section V of Exhibit 1 or Section III of Exhibit 2. These sections are not applicable to acquired software requiring no development or modification, unless acquired design documentation is incomplete.

5.5.2 Design documentation for each software class shall include at a minimum the following:

o engineering/scientific software - mathematical models and numerical methods description, and user's manual. The specific items selected for inclusion are so indicated in Section V of the Software Requirements Form, Engineering/Scientific Software (Exhibit 1).

o support software - user's manual. The specific items selected for inclusion are so indicated in Section III of the Software Requirements Form, Support Software (Exhibit 2).

o system maintained software - available documentation for the particular software version.

5.5.3 The design documentation necessary to the FIDR (as determined by the project manager and sponsor requirements) shall be completed prior to the FIDR. Delayed design documentation shall be specified in Section VII, Software Requirements Form, Exhibit 1 and shall be added by following SCP-314, Software Configuration Management.

5.6 SCPs Required for Engineering/Scientific Software

5.6.1 Engineering/scientific software shall be acquired in accordance with SCP-317, Transfer of Software, Data and/or Documentation; or PAP-401, Preparation, Review and Approval of Purchase Requisitions.

5.6.2 When development or modification is required for engineering/ scientific software and/or design documentation, an FIDR shall be performed in accordance with SCP-313, Final Internal Development Review of Software and Development.

5.6.3 After completion of an FIDR, or after transfer if no modification or development occurs, all engineering/scientific software shall be configuration managed in accordance with SCP-314, Software Configuration Management.

5.6.4 All engineering/scientific software shall be tested in accordance with SCP-315, Conversion Testing, Verification, and/or Validation of Software, as follows:

o all acquired engineering/scientific software shall be conversion tested

o all engineering/scientific software shall be verified. Verification shall be independent if prescribed in research project planning documents.

o validation of engineering/scientific software shall be performed if prescribed in research project planning documents.

NOTE: SCP-315, Conversion Testing, Verification, and/or Validation of Software, may Le implemented only after placing software under configuration management.

5.6.5 Use of engineering/scientific software shall be documented in accordance with SCP-316, Software Application Control.

5.7 SCPs Required for Support Software

5.7.1 Support software shall be acquired in accordance with SCP-317, Transfer of Software, Data and/or Documentation; or PAP-401, Preparation, Review and Approval of Purchase Requisitions.

5.7.2 If support software is used repeatedly or by multiple users, it shall require configuration management in accordance with SCP-314, Software Configuration Management.

5.7.3 If support software is acquired, it shall be conversion tested in accordance with SCP-315, Conversion Testing, Verification and/or Validation of Software. If it is developed or modified, support software shall be verified in accordance with SCP-315, Conversion Testing, Verification, and/or Validation of Software.

5.7.4 Streams of commands (or sequences of streams of commands) used to execute system maintained software to generate reportable results shall be classed as support software and shall be verified in accordance with SCP-315, Conversion Testing, Verification, and/or Validation.

5.7.5 Use of support software shall be documented in accordance with SCP-316, Software Application Control, unless documented as part of an application run of engineering/scientific software.

5.8 SCPs Required for System Maintained Software

5.8.1 If system maintained software is acquired, it shall be acquired in accordance with SCP-317, Transfer of Software, Data and/or Documentation; or PAP-401, Preparation, Review and Approval of Purchase Requisitions. If system maintained software has been obtained previously, transfer of software and documentation to a

research project using SCP-317, Transfer of Software, Data and/or Documentation, is not required.

5.8.2 The preparer of the Software Requirements Form, System Maintained Software (Exhibit 3), shall assure that the name, supplier, version and installation date of system maintained software (e.g., commercial software such as LOTUS 1-2-3, RS/1, SAS, or DISSPLA) are specified on the form.

5.8.3 Use of system maintained software shall be documented in accordance with SCP-316, Software Application Control.

5.9 SCPs Required for Data

5.9.1 As shown in Exhibit 4, data shall be acquired from another research project at PNL or from outside PNL in accordance with SCP-317, Transfer of Software, Data and/or Documentation; PAP-401, Preparation, Review and Approval of Purchase Requisitions; or from open literature.

5.9.2 Data used in software development, testing or application shall be verified and configuration managed in accordance with SCP-318, Control of Data Bases.

5.9.3 The preparer of the Software Requirements Form (Exhibit 1, 2 or 3) shall indicate the name of the data base steward appointed in accordance with PAP-205, Quality Assurance Plans, in

o   item 10, Section II of Exhibit 1, for engineering/scientific software

o   item 10, Section II of Exhibit 2, for support software

o   item 9, Section II of Exhibit 3, for system maintained software.

5.10 Changing an Approved Software Requirements Form

After a Software Requirements Form has been approved by an ITR and by the sponsor, if required, to make changes in software class, design input, design documentation requirements, testing options, or required SCPs, the project manager shall assure that a revised Software Requirements Form (Exhibit 1, 2 or 3) is issued in accordance with this SCP and receives the same level of approval as the original Software Requirements Form. A revised Software Requirements Form shall be indicated by sequentially assigning a revision number, where the initial Software Requirements Form shall be designated Revision no. 0.

5.11 <u>Reporting of Deficiencies</u>

If the Software Requirements Form (Exhibit 1, 2 or 3) is determined to be deficient after being submitted as a research project record, the project manager shall evaluate and document the deficiency in accordance with PAP-1502, Controlling Deviations from QA Requirements and Established Procedures.

6.0 <u>REFERENCED DOCUMENTS</u>

6.1 Exhibit 1, Software Requirements Form, Engineering/Scientific Software

6.2 Exhibit 2, Software Requirements Form, Support Software

6.3 Exhibit 3, Software Requirements Form, System Maintained Software

6.4 Exhibit 4, Flowchart for Input Data to Software

6.5 SCP-313, Final Internal Development Review of Software and Documentation

6.6 SCP-314, Software Configuration Management

6.7 SCP-315, Conversion Testing, Verification and/or Validation of Software

6.8 SCP-316, Software Application Control

6.9 SCP-317, Transfer of Software, Data and/or Documentation

6.10 SCP-318, Control of Data Bases

6.11 PAP-101, Communication and Commitment (Interface) Control

6.12 PAP-205, Quality Assurance Plans

6.13 PAP-401, Preparation, Review and Approval of Purchase Requisitions

6.14 PAP-604, Independent Technical Review

6.15 PAP-1502, Controlling Deviations from QA Requirements and Established Procedures

6.16 IEEE-std-730-1984, <u>Standard for Software Quality Assurance Plans</u>

6.17 ANSI/ASME NQA-1-1983, <u>Quality Assurance Program Requirements for Nuclear Power Plants</u>, Supplement 3S-1, Supplementary Requirements for Design Control

6.18 DOD-STD-480A, Notice 1, <u>Configuration Control-Engineering Changes, Deviations and Waivers</u>

6.19 NUREG-0856, <u>Final Technical Position on Documentation of Codes for High-Level Waste Management</u>

SOFTWARE REQUIREMENTS FORM
ENGINEERING/SCIENTIFIC SOFTWARE
Revision no. _____

(Answer every question or, if appropriate for questions with a line in front of the question number, specify not applicable, N/A. If not otherwise specified, proceed to the next question in sequence.)

1) Software name (and version, if applicable) _____
2) Name of code custodian _____
3) Project title _____ Project no. _____
4) Function of this engineering/scientific software:

SECTION I: Determination of Required Software Control Procedures *(See p. 6 of this exhibit for a flowchart of the sequence.)*

5) Is this engineering/scientific software going to be acquired from outside PNL or from another PNL research project (with a different project number)?
___ yes
___ no *(Mark N/A on questions 6-8. Go to question 9.)*

___ 6) Identify software origin and version(s):

___ 7) Describe available software documentation and version(s):

___ 8) Will this engineering/scientific software and/or its design documentation be modified at PNL?

    ___ yes   Required SCPs (in addition to SCP-312, Determination of Software Requirements) in order of application:

        SCP-317, Transfer of Software, Data and/or Documentation *(to transfer software and its documentation)*

        SCP-313, Final Internal Development Review of Software and Documentation

        SCP-314, Software Configuration Management

        SCP-315, Conversion Testing, Verification, and/or Validation of Software

        SCP-316, Software Application Control

    (Indicate required SCPs in Section VI. Mark N/A on question 9. Go to question 10.)

Revision no. _____
Software name and version _____
Project name and no. _____

SCP-312, Rev. 1
EXHIBIT 1
Page 2 of 6

___ no   Required SCPs (in addition to SCP-312, Determination of
Software Requirements) in order of application:

SCP-317, Transfer of Software, Data and/or Documentation *(to
transfer software and its documentation)*
SCP-314, Software Configuration Management
SCP-315, Conversion Testing, Verification, and/or Validation of
Software
SCP-316, Software Application Control

*(Indicate required SCPs in Section VI. Mark N/A on question 9. Go
to question 10.)*

___ 9) Is this engineering/scientific software and/or its design documentation going
to be developed at PNL?

___ yes   Required SCPs (in addition to SCP-312, Determination of
Software Requirements) in order of application:

SCP-313, Final Internal Development Review of Software and
Documentation
SCP-314, Software Configuration Management
SCP-315, Conversion Testing, Verification, and/or Validation of
Software
SCP-316, Software Application Control

(Indicate required SCPs in Section VI.)

___ no   *(You have answered "no" to questions 5 and 9. You need to
reevaluate whether you have chosen the correct software class. If
the class is correct, either question 5 or question 9 must be
answered by "yes.")*

SECTION II: Data Associated with this Engineering/Scientific Software

10) Will data or data base(s) be used to determine input for this
engineering/scientific software?
___ yes   The data base steward designated in accordance with PAP-205,
Quality Assurance Plans, is: _____
___ no   *(Mark N/A on questions 11-13. Go to question 14.)*

__11) Will data or data base(s) be acquired from outside PNL or from another PNL
research project (with a different project number)?

___ yes   Required SCPs (in addition to those in Section I) in order
of application:

SCP-317, Transfer of Software, Data and/or Documentation *(to
transfer data and its documentation)*
SCP-318, Control of Data Bases

(Indicate required SCPs in Section VI.)

___ no   Required SCP (in addition to those in Section I):

SCP-318, Control of Data Bases

*(Indicate required SCP in Section VI. Mark N/A on questions 11 and
13. Go to question 14.)*

Revision no. _____
Software name and version _____
Project name and no. _____

SCP-312, Rev. 1
EXHIBIT 1
Page 3 of 6

__12) Identify origin of data or data base(s):

__13) Describe available documentation of data or data base(s) including version(s):

SECTION III: Determination of Options for SCP-315, Conversion Testing, Verification, and/or Validation of Software

__14) Is conversion testing required for this acquired engineering/scientific software? *(Applicable if engineering/scientific software is acquired from outside PNL or being installed on a different computer or different operating system.)*
    ___ yes *(Indicate so under Section VI.)*
    ___ no

15) Is independent verification required?
    ___ yes *(Indicate so under Section VI.)*

    ___ "Independent" is defined to be verification by competent PNL individual(s) other than those from whom the work originated (they may be users, but they shall not have designed or developed the software).

    ___ "Independent" is defined to be verification by competent individual(s) outside PNL.

    ___ "Independent" is defined to be verification by:

    ___ no

16) Is validation required?
    ___ yes *(Indicate so under Section VI.)*
    ___ no

__17) Is sponsor approval of testing plans required?
    ___ yes Specify particular type(s) of testing for which approval is required:

    ___ no

Revision no. _____
Software name and version _____
Project name and no. _____

SCP-312, Rev. 1
EXHIBIT 1
Page 4 of 6

SECTION IV. Design Input *(applicable if answer to question 8 or 9 is "yes")*

__18) List research project planning documents (title and page numbers) containing design input (see Section 5.4.1):

__19) Is additional design input attached (see Section 5.4.2)?
___ yes
___ no
Explain any exclusions of items in Section 5.4.2:

SECTION V: Design Documentation *(applicable if answer to question 8 or 9 is "yes". Mark all items to be included.)*

20) Mathematical models and numerical methods descriptions shall include:
___ Design input (i.e., documentation of items 18 and 19)
___ Statement and description of the problem
___ Applicable assumptions and limitations (e.g., appropriateness of algorithms)
___ Numerical techniques/methods
___ Relevant discretized (or otherwise transformed numerical solution) equations and derivations
___ Numerical stability and accuracy of methods
___ Notation for variables and equations
___ Important computational characteristics
___ References and sources
___ Other _____

21) User's manual shall include:
___ Hardware requirements including computer type and operating system
___ Software listing (handwritten, computer generated or on microfiche)
___ Testing documentation relevant to Final Internal Development Review
___ Structure and organization of the software by flowchart, software design language, or other appropriate means
___ Data input and output information
___ Model and system interfaces
___ Coding standards
___ Sample and/or test problems
___ Input/output requirements (e.g., libraries and compilers)
___ Other _____

__22) Do research project planning documents or other contractual documents require compliance to a specific standard?

___ yes   Identify the standard:

___ no

Revision no. _____
Software name and version _____
Project name and no. _____

SCP-312, Rev. 1
EXHIBIT 1
Page 5 of 6

SECTION VI: Summary of Required SCPs

        __X__ SCP-312, Determination of Software Requirements
        _____ SCP-313, Final Internal Development Review of Software and Documentation
        __X__ SCP-314, Software Configuration Management
        __X__ SCP-315, Conversion Testing, Verification, and/or Validation of Software
                _____ conversion testing
                __X__ verification (independent? ___ yes ___ no)
                _____ validation
        __X__ SCP-316, Software Application Control
        _____ SCP-317, Transfer of Software, Data and/or Documentation
                _____ software
                _____ documentation of software
                _____ data
                _____ documentation of data
        _____ SCP-318, Control of Data Bases

SECTION VII: Additions or Exceptions to Questions 1-22   *(attach additional pages if necessary)*

__23) Describe any additions or exceptions to the above:

__24) Provide explanations for additions or exceptions:

SECTION VIII: Approvals of Answers to Questions 1-24

  25) Prepared by:

_____
Signature   Date

  26) Approved in accordance with PAP-604, Independent Technical Review, by:

_____
Project Manager or Independent Technical   Date
Reviewer (if not project manager)

  27) Reviewed by:

_____
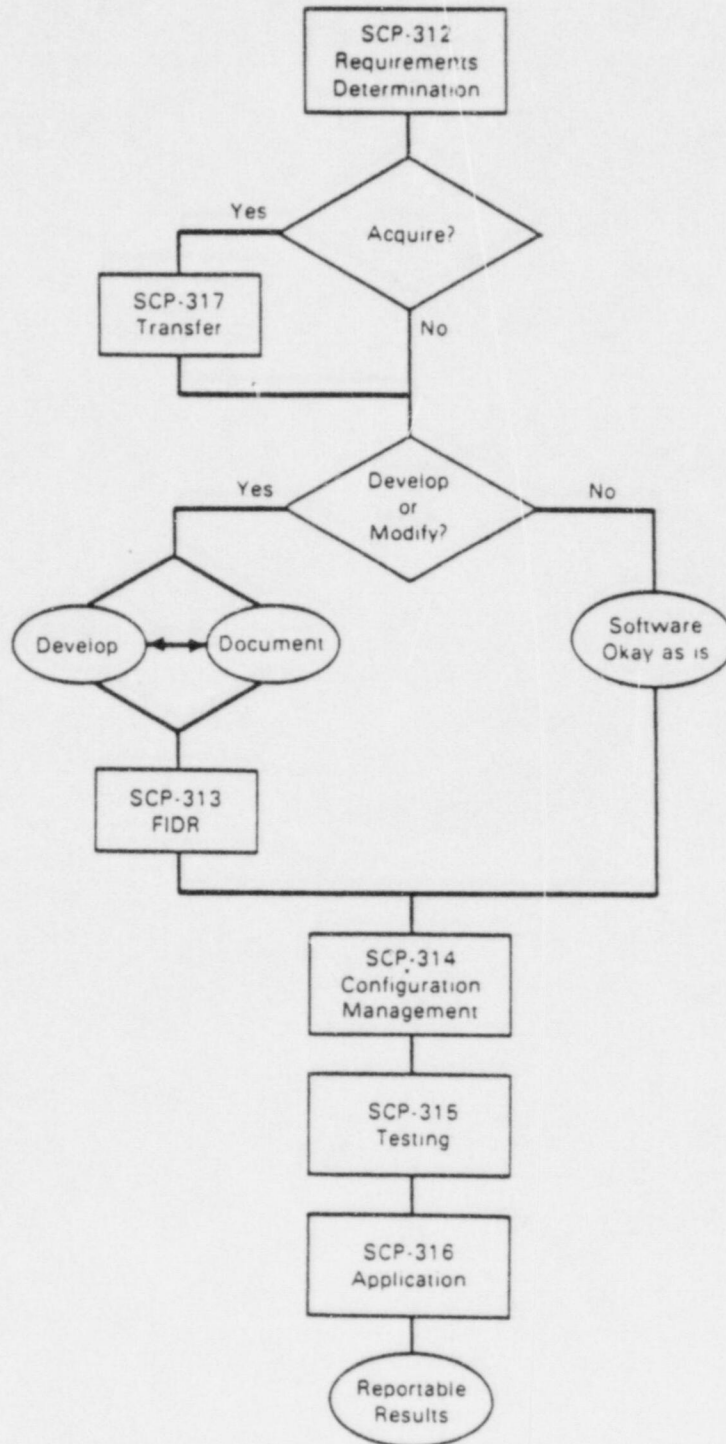QAD Representative   Date

__28) Concurrence by:

_____
Sponsor Representative (if required by   Date
research project planning documents)

Engineering/Scientific Software

SOFTWARE REQUIREMENTS FORM
SUPPORT SOFTWARE
Revision no. _____

(Answer every question, or if appropriate for a question with a line in front of the question number, specify not applicable, N/A. If not otherwise specified, proceed to the next question in sequence.)

1) Software name (and version, if applicable) _____
2) Name of code custodian _____
3) Project title _____   Project no. _____
4) Function of this support software:

5) Is this support software a stream of commands used to execute system maintained software?
   ___ yes
   ___ no

SECTION I: Determination of Required Software Control Procedures *(See p. 4 of this exhibit for a flowchart of the sequence.)*

6) Will this support software receive repeated use or be used by multiple users?

   ___ yes   Required SCP (in addition to SCP-312, Determination of Software Requirements):

             SCP-314, Software Configuration Management

             (Indicate required SCP in Section IV.)

   ___ no

7) Is this support software going to be acquired from outside PNL or from another PNL research project (with a different project number)?

   ___ yes   Required SCPs (in addition to SCP-312, Determination of Software Requirements) in order of application *(apply SCP-314 before SCP-315, if SCP-314 is required from question 6)*:

             SCP-317, Transfer of Software, Data and/or Documentation *(to transfer software and its documentation)*
             SCP-315, Conversion Testing, Verification, and/or Validation of Software
             SCP-316, Software Application Control

             (Indicate required SCPs in Section IV.)

   ___ no   Required SCPs (in addition to SCP-312, Determination of Software Requirements) in order of application *(apply SCP-314 first, if SCP-314 is required from question 6)*:

             SCP-315, Conversion Testing, Verification, and/or Validation of Software
             SCP-316, Software Application Control

             (Indicate required SCPs in Section IV. Mark N/A on questions 8 and 9. Go to question 10.)

Revision no. _____
Software name and version _____
Project name and no. _____

SCP-312, Rev. 1
EXHIBIT 2
Page 2 of 4

__ 8) Identify software origin and version(s):

__ 9) Describe available software documentation and version(s):

SECTION II: Data Associated with this Support Software

10) Will data or data base(s) be used to determine input for this support
software?

   ___ yes The data base steward designated in accordance with PAP-205,
             Quality Assurance Plans, is: _____

   ___ no   *(Mark N/A on questions 11-13. Go to question 14.)*

__11) Will data or data base(s) be acquired from outside PNL or from another PNL
research project (with a different project number)?

   ___ yes Required SCPs (in addition to those in Section I) in order
           of application:

           SCP-317, Transfer of Software, Data and/or Documentation *(to
                  transfer data and its documentation)*
           SCP-318, Control of Data Bases

           (Indicate required SCPs in Section IV.)

   ___ no   Required SCP (in addition to those in Section I):

           SCP-318, Control of Data Bases

           *(Indicate the required SCP in Section IV. Mark N/A on questions 12
           and 13. Go to question 14.)*

__12) Identify origin of data or data base(s):

__13) Describe available documentation of data or data base and version(s):

SECTION III: Design Documentation *(Mark all items to be included.)*

14) User's manual shall include:

   ___ Hardware requirements including computer type and operating system
   ___ Software listing (handwritten, computer generated or on microfiche)
   ___ Testing documentation
   ___ Structure and organization of the software by flowchart, software
        design language, or other appropriate means
   ___ Data input and/or output information
   ___ Model and system interfaces
   ___ Coding standards
   ___ Sample and/or test problems
   ___ Input/output requirements (e.g., libraries and compilers)
   ___ Other _____

Revision no. _____   SCP-312, Rev. 1
Software name and version _____   EXHIBIT 2
Project name and no. _____   Page 3 of 4

SECTION IV: Summary of Required SCPs

    X SCP-312, Determination of Software Requirements
       SCP-314, Software Configuration Management
    X SCP-315, Conversion Testing, Verification, and/or Validation of Software
            conversion testing
      X verification
            validation
    X SCP-316, Software Application Control
       SCP-317, Transfer of Software, Data and/or Documentation
            software
            documentation of software
            data
            documentation of data
       SCP-318, Control of Data Bases

SECTION V: Additions or Exceptions to Questions 1-14 *(attach additional pages if necessary)*

__15) Describe any additions or exceptions to the above:

__16) Provide explanations for additions or exceptions:

SECTION VI: Approvals of Answers to Questions 1-16

   17) Prepared by:

_____

Signature   Date

   18) Approved in accordance with PAP-604, Independent Technical Review, by:

_____

Project Manager or Independent Technical   Date
Reviewer (if not project manager)

Support Software

```
          ┌─────────────────┐
          │     SCP-312     │
          │  Requirements   │
          │  Determination  │
          └─────────────────┘
                   │
                ◇ Acquire? ◇
   Yes ─────────┘        │ No
    │                    │
┌─────────┐              │
│ SCP-317 │              │
│ Transfer│              │
└─────────┘              │
    └────────────────────┤
                         │
                ◇ Repeated       ◇
         Yes    Use or Multiple   No
          ┌─── Users?          ───┐
          │                       │
  ┌──────────────┐        ┌──────────────┐
  │   SCP-314    │◄- - - -│ Configuration│
  │ Configuration│        │  Management  │
  │  Management  │        │  is Optional │
  └──────────────┘        └──────────────┘
          └───────────────────────┘
                     │
               ┌──────────┐
               │ SCP-315  │
               │ Testing  │
               └──────────┘
                     │
               ┌──────────┐
               │ SCP-316  │
               │Application│
               └──────────┘
                     │
               ╭──────────╮
               │ Support  │
               │of E/S or System│
               │ Maintained│
               │ Software │
               ╰──────────╯
```
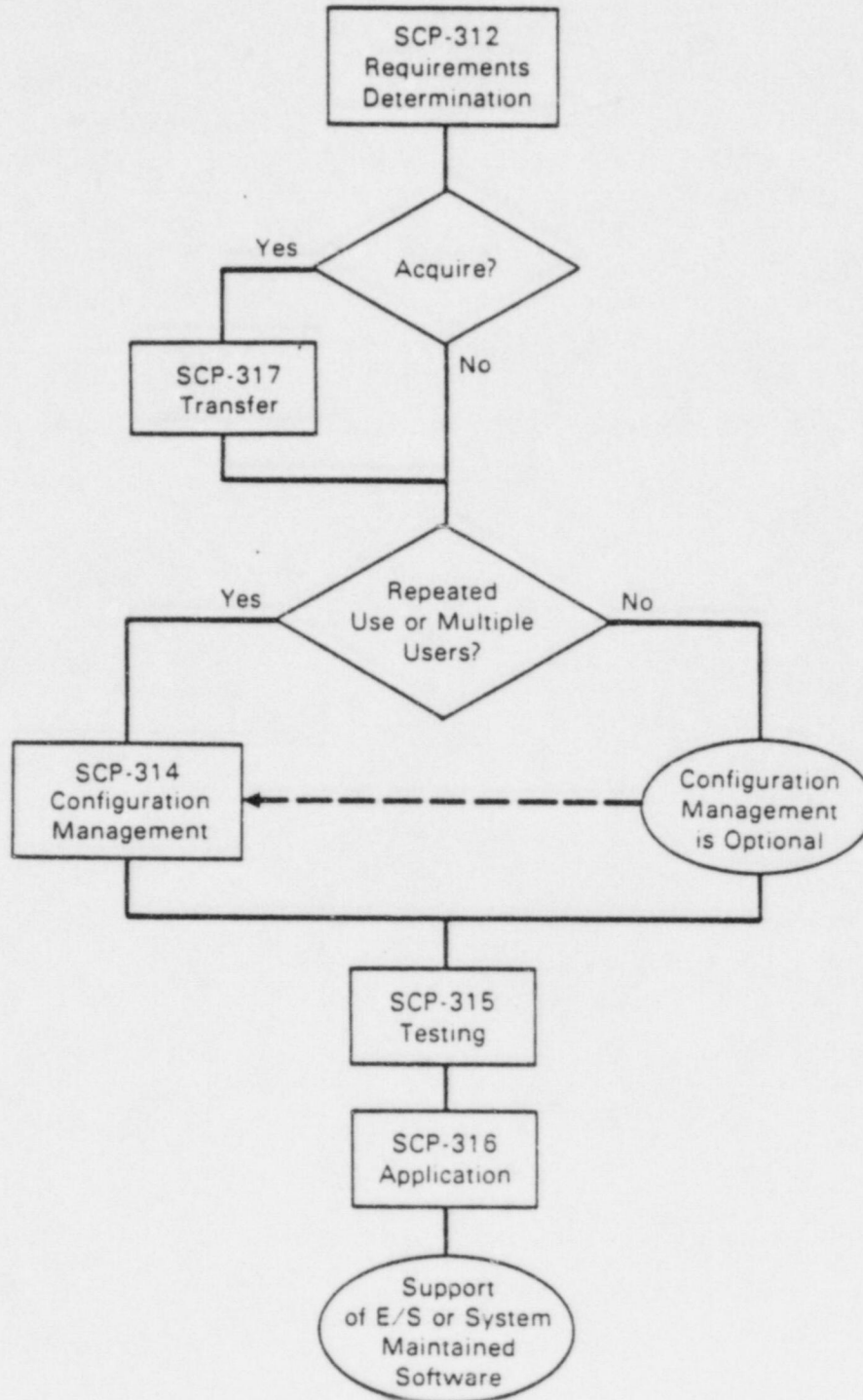
SOFTWARE REQUIREMENTS FORM
SYSTEM MAINTAINED SOFTWARE
Revision no. _____

(Answer every question or, if appropriate for a question with a line in front of the question number, specify not applicable, N/A. If not otherwise specified, proceed to the next question in sequence.)

1) Software name (and version, if applicable) _____
2) Name of code custodian _____
3) Project title _____ Project no. _____
4) Function of this system maintained software:

5) Does this system maintained software require a stream of commands to execute various options?
   ___ yes *(Class each stream of commands as support software in accordance with Section 5.7.4.)*
   ___ no Explain:

SECTION I: Determination of Required Software Control Procedures *(See p. 4 of this exhibit for a flowchart of the sequence.)*

6) Is this system maintained software to be purchased or transferred?
   ___ Purchased. *(See PAP-401, Preparation, Review and Approval of Purchase Requisitions.)*

   ___ Transferred. Required SCP (in addition to SCP-312, Determination of Software Requirements):

   SCP-317, Transfer of Software, Data and/or Documentatio￼ *(t. transfer software and its documentation)*

   (Indicate required SCP in Section III.)

   ___ Neither. (Software is already installed from previous purchase or transfer.)

7) Identify origin (vendor) of software and version(s):

8) Describe available software documentation and version(s):

Revision no. _____
Software name and version _____
Project name and no. _____

SCP-312, Rev. 1
EXHIBIT 3
Page 2 of 4

SECTION II: Data Associated with this System Maintained Software

9) Will data or data base(s) be used to determine input for this system maintained software?

   ___ yes   The data base steward designated in accordance with PAP-205, Quality Assurance Plans, is: _____

   ___ no   *(Mark N/A to questions 12-14. Go to question 15.)*

_10) Will data or data base(s) be acquired from outside PNL or from another PNL research project (with a different project number)?

   ___ yes   Required SCP (in addition to those in Section I):

             SCP-316, Software Application Control
             SCP-317, Transfer of Software, Data and/or Documentation *(to transfer data and its documentation)*
             SCP-318, Control of Data Bases

             (Indicate required SCPs in Section III.)

   ___ no   Required SCP Required SCP (in addition to those in Section I):

             SCP-316, Software Application Control
             SCP-318, Control of Data Bases

             (Indicate required SCPs in Section III. Mark N/A on questions 13 and 14. Go to question 15.)

_11) Identify origin of data or data base(s):

_12) Describe available documentation of data or data base(s) and version(s):

SECTION III: Summary of Required SCPs

   __X__ SCP-312, Determination of Software Requirements
   __X__ SCP-316, Software Application Control
   _____ SCP-317, Transfer of Software, Data and/or Documentation
         _____ software
         _____ documentation of software
         _____ data
         _____ documentation of data
   _____ SCP-318, Control of Data Bases

Revision no. _____   SCP-312, Rev. 1
Software name and version _____   EXHIBIT 3
Project name and no. _____   Page 3 of 4

SECTION IV: Additions and Exceptions to Questions 1-12 *(attach additional pages if necessary)*

__13) Describe any additions or exceptions to the above:

__14) Provide explanations for additions or exceptions:

SECTION V: Approvals of Answers .to Questions 1-14

15) Prepared by:

_____
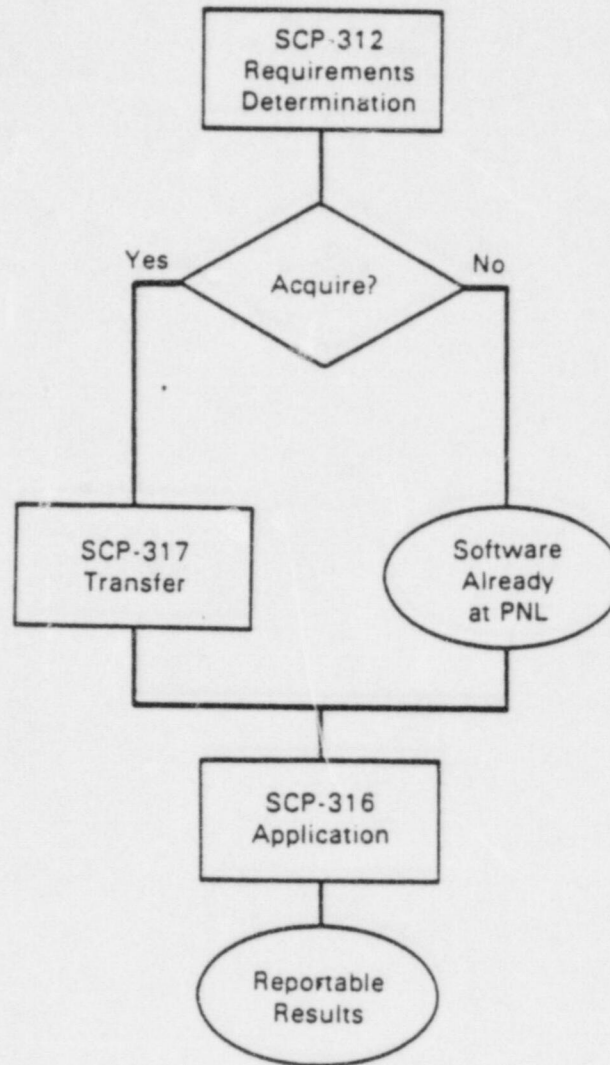Signature   Date

16) Approved in accordance with PAP-604, Independent Technical Review, by:

_____
Project Manager or Independent Technical   Date
Reviewer (if not project manager)

System Maintained Software

Input Data to Software
(to development, testing or application)

```
        ┌─────────────────┐
        │ SCP-312         │
        │ Requirements    │
        │ Determination   │
        └─────────────────┘
                │
              ◇ Data              No      ╭──────────────╮
                Required ? ──────────────▶│ No Further   │
              ◇                           │ SCPs Required│
                │ Yes                     ╰──────────────╯
        ┌───────┴────────┐
   ┌──────────┐    ╭──────────────╮
   │ SCP-317  │    │ Data in Open │
   │ Transfer │    │ Literature   │
   └──────────┘    │ or in Project│
        │          ╰──────────────╯
        └───────┬────────┘
        ┌─────────────────┐
        │ SCP-318         │
        │ Control of Data │
        └─────────────────┘
                │
          ╭──────────────╮
          │ Data Input   │
          │ to Software  │
          ╰──────────────╯
```