NUREG/CR-6316
SAIC-95/1028
Vol. 6

Guidelines for the Verification and Validation of Expert System Software and Conventional Software

Validation Scenarios

Prepared
S. M. Mirsky, J. E. Hayes, L. A. Miller

RECEIVED
APR 2 1 1995
OSTI

Science Applications International Corporation

Prepared for
U.S. Nuclear Regulatory Commission

and

Electric Power Research Institute

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

AVAILABILITY NOTICE

Availability of Reference Materials Cited in NRC Publications

Most documents cited in NRC publications will be available from one of the following sources:

1. The NRC Public Document Room, 2120 L Street, NW., Lower Level, Washington, DC 20555-0001
2. The Superintendent of Documents, U.S. Government Printing Office, P. O. Box 37082, Washington, DC 20402-9328
3. The National Technical Information Service, Springfield, VA 22161-0002

Although the listing that follows represents the majority of documents cited in NRC publications, it is not intended to be exhaustive.

Referenced documents available for inspection and copying for a fee from the NRC Public Document Room include NRC correspondence and internal NRC memoranda; NRC bulletins, circulars, information notices, inspection and investigation notices; licensee event reports; vendor reports and correspondence; Commission papers; and applicant and licensee documents and correspondence.

The following documents in the NUREG series are available for purchase from the Government Printing Office: formal NRC staff and contractor reports, NRC-sponsored conference proceedings, international agreement reports, grantee reports, and NRC booklets and brochures. Also available are regulatory guides, NRC regulations in the *Code of Federal Regulations,* and *Nuclear Regulatory Commission Issuances.*

Documents available from the National Technical Information Service include NUREG-series reports and technical reports prepared by other Federal agencies and reports prepared by the Atomic Energy Commission, forerunner agency to the Nuclear Regulatory Commission.

Documents available from public and special technical libraries include all open literature items, such as books, journal articles, and transactions. *Federal Register* notices, Federal and State legislation, and congressional reports can usually be obtained from these libraries.

Documents such as theses, dissertations, foreign reports and translations, and non-NRC conference proceedings are available for purchase from the organization sponsoring the publication cited.

Single copies of NRC draft reports are available free, to the extent of supply, upon written request to the Office of Administration, Distribution and Mail Services Section, U.S. Nuclear Regulatory Commission, Washington, DC 20555-0001.

Copies of industry codes and standards used in a substantive manner in the NRC regulatory process are maintained at the NRC Library, Two White Flint North, 11545 Rockville Pike, Rockville, MD 20852-2738, for use by the public. Codes and standards are usually copyrighted and may be purchased from the originating organization or, if they are American National Standards, from the American National Standards Institute, 1430 Broadway, New York, NY 10018-3308.

DISCLAIMER NOTICE

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for any third party's use, or the results of such use, of any information, apparatus, product, or process disclosed in this report, or represents that its use by such third party would not infringe privately owned rights.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

NUREG/CR-6316
SAIC-95/1028
Vol. 6

Guidelines for the Verification and Validation of Expert System Software and Conventional Software

Validation Scenarios

Manuscript Completed: February 1995
Date Published: March 1995

Prepared by
S. M. Mirsky, J. E. Hayes, L. A. Miller

Science Applications International Corporation
1710 Goodridge Drive
McLean, VA 22102

Prepared for
Division of Systems Technology
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, DC 20555-0001
NRC Job Code L1530

and

Nuclear Power Division
Electric Power Research Institute
3412 Hillview Avenue
Palo Alto, CA 94303

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

MASTER

ABSTRACT

This report is the sixth volume in a series of reports describing the results of the Expert System Verification and Validation (V&V) project which is jointly funded by the US Nuclear Regulatory Commission and the Electric Power Research Institute. The ultimate objective is the formulation of guidelines for the V&V of expert systems for use in nuclear power applications. This activity was concerned with the development of a methodology for selecting validation scenarios and subsequently applying it to two expert systems used for nuclear utility applications.

Validation scenarios were defined and classified into five categories: PLANT, TEST, BASICS, CODE, and LICENSING. A sixth type, REGRESSION, is a composite of the others and refers to the practice of using trusted scenarios to ensure that modifications to software did not change unmodified functions. Rationale was developed for preferring scenarios selected from the categories in the order listed and for determining under what conditions to select scenarios from other types.

A procedure incorporating all of the recommendations was developed as a generalized method for generating validation scenarios. The procedure was subsequently applied to two expert systems used in the nuclear industry and was found to be effective, given that an experienced nuclear engineer made the final scenario selections. A method for generating scenarios directly from the knowledge base component was suggested.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

EXECUTIVE SUMMARY

This report is the sixth volume in a series of reports describing the results from the Expert System Verification and Validation (V&V) project which is jointly funded by the United States Nuclear Regulatory Commission (USNRC) and the Electric Power Research Institute (EPRI). The ultimate objective of this project is the formulation and documentation of guidelines for V&V of Artificial Intelligence systems in the nuclear power industry.

This report is concerned with the development of a methodology for selecting validation scenarios. The task encompasses an extensive assessment of the current state of the art in developing validation scenarios for nuclear power industry-related computer software. This combined telephone and literature assessment included nuclear industry, consultant, architect-engineer, nuclear steam supply system, and national laboratory organizations and resulted in validation scenario data for 39 specific instances of nuclear software validations.

From the assessment, validation scenarios were classified into one of six types which are denoted as: PLANT, TEST, BASICS, CODE, LICENSING (scenarios associated with regulatory concerns), and REGRESSION. These scenario types were rank ordered by importance, as above, in terms of their application to new or modified software. A set of basic rules was developed governing the selection of type and number of validation scenarios which is related to V&V Class and other factors such as the availability and pedigree of nuclear power plant or experimental test facility data. Finally, guidelines were developed to assist a user in selecting the number and type of validation scenarios.

A summary of what was learned in this task consists of:

1. There are no standards, regulations, or guidelines for the selection of number and type of validation scenarios in the nuclear industry.

2. Six basic types of validation scenarios may be identified: PLANT, TEST, BASICS, CODE, LICENSING, and REGRESSION, where the last is a composite of the others and used to determine if the latest modifications to a software program affected other aspects than the parts modified.

3. The types of validation scenarios used depend on their availability, applicability, fidelity, and order of importance. Rules based on these factors were developed to provide guidance for choosing scenario types.

4. The number of validation scenarios used can vary greatly, depending on the V&V Class, whether the software is new or just a modification of already approved software, and what its range of application is (i.e., very specific vs. very general). Rules based on these factors were developed to guide determination of the number of scenarios needed.

5. The validation scenario selection rules developed in 3) and 4) above were applied to two AI systems used in the nuclear industry and were found to be effective. However, an experienced nuclear engineer conversant with transient event testing and safety standards was required to make the final decision concerning specific scenario cases.

6. The validation scenario guidelines apply both to AI systems and conventional software because they test the functional capability of the system without regard for its structure or method of development.

7. One method for generating scenarios directly from the Knowledge-Base component of AI systems was suggested. (This method has not yet been tried, however.)

1 INTRODUCTION

This report is the sixth volume in a series of task reports describing the results from the Expert System Verification and Validation (V&V) project. The ultimate objective of this project is the formulation of guidelines for the V&V of AI systems in the nuclear power industry. This work is jointly sponsored by the United States Nuclear Regulatory Commission (USNRC) and the Electric Power Research Institute (EPRI).

1.1 Purpose and Scope

The purpose of the activity described in this report is to develop a method to generate validation scenarios and generate a set of validation scenarios that will evaluate and challenge the functional requirements for the selected systems. It should be noted that validation scenarios are used after the basic V&V testing as presented in Volume 7 (Miller, et al.) of the system is completed. Validation scenario execution augments the basic V&V process. The two systems that are being used for this project are denoted System A and B and are both direct applications for nuclear power plants. System A is designed to assist boiling water reactor (BWR) operators in the selection and use of appropriate emergency operating procedures (EOPs) during any abnormal event or transient. SAIC used an early generic version of System A for this project. A plant specific, verified and validated version of System A (Chang, Cheng, 1990) has been installed at the Kuosheng nuclear power plant in Taiwan. System B is designed for Pressurized Water Reactors (PWRs) and is in use at the USNRC emergency operations center (EOC). System B provides guidance to the USNRC reactor safety team at the EOC during a nuclear power plant event. This guidance is in the form of status of critical safety functions at the plant.

This work includes an extensive assessment of the current state of the art in the generation and use of validation scenarios in the nuclear power industry. Nuclear power utilities, architect-engineers, nuclear steam supply system (NSSS) vendors, consultants, and national laboratories were all contacted to determine the state-of-the-art in selecting validation scenarios for nuclear power software. The information gathered formed the basis for developing a general method of selecting validation scenarios for AI systems and it was then used to develop specific scenarios for Systems A and B. This work does not include the development of validation scenarios which are designed to include recovery from software faults. The V&V Guidelines in Volumes 5 and 7 include such techniques as robustness testing and functional testing which examine fault tolerance.

1.2 Relationship to Other Project Activities

The work described in this report, identified as Activity 8 on Figure 1.2-1, is independent of most of the previous tasks in that it deals with final system validation alone, not with verification or certification, nor with other life cycle phases, which are the key aspects of Activities 3, 4, 5, 6, and 7. However, as shown in Figure 1.2-1, the results of these earlier tasks have influenced this task by virtue of their input to Activity 7 and its assessment of method effectiveness.

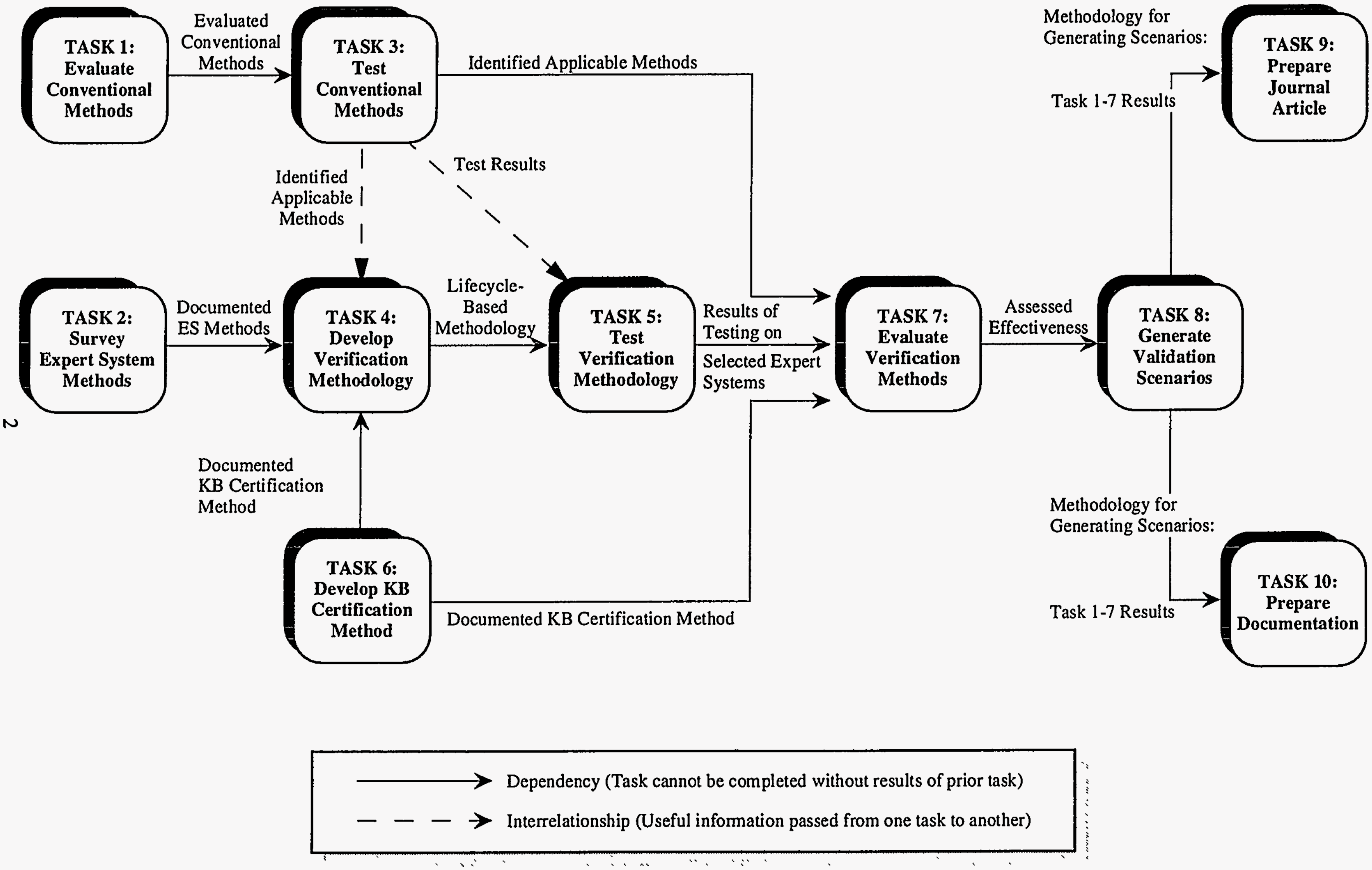

TASK 1: Evaluate Conventional Methods
Evaluated Conventional Methods
TASK 3: Test Conventional Methods
Identified Applicable Methods
Methodology for Generating Scenarios:
Task 1-7 Results
TASK 9: Prepare Journal Article
Identified Applicable Methods
Test Results
TASK 2: Survey Expert System Methods
Documented ES Methods
TASK 4: Develop Verification Methodology
Lifecycle-Based Methodology
TASK 5: Test Verification Methodology
Results of Testing on Selected Expert Systems
TASK 7: Evaluate Verification Methods
Assessed Effectiveness
TASK 8: Generate Validation Scenarios
Documented KB Certification Method
Methodology for Generating Scenarios:
Task 1-7 Results
TASK 10: Prepare Documentation
TASK 6: Develop KB Certification Method
Documented KB Certification Method
Dependency (Task cannot be completed without results of prior task)
Interrelationship (Useful information passed from one task to another)

Figure 1.2-1 TASK DEPENDENCIES AND INTERRELATIONSHIPS

1.3 Report Organization

Section 2 describes the information gathered on current validation scenario philosophy in the nuclear industry. This information is analyzed in Section 3 and the resulting methodology for validation scenario development is presented in Section 4. The effectiveness of validation scenarios was not numerically rated as in other activities of this project because the source of data (i.e., discussions with nuclear industry and review of nuclear industry validation scenario documentation) provides qualitative assertions about the acceptability of a specific set of validation scenarios. For software that has been accepted by the USNRC for use with nuclear power plants, the validation scenarios were presumed to be acceptable. The cost effectiveness and efficiency of any given scenario set is subject to engineering judgment. Section 5 presents the validation scenarios selected for Systems A and B in accordance with the guidelines presented in Section 4. Section 6 provides a conceptual description of an alternative means for generating validation scenarios directly from the knowledge base component of AI systems. Finally, Section 7 presents overall results and conclusions. A list of references is provided in Section 8.

2 NUCLEAR INDUSTRY VALIDATION SCENARIO ASSESSMENT

The strategy adopted for developing a methodology for generating validation scenarios was as follows:

1) Conduct an assessment of organizations in the nuclear power industry which use production software, asking participants what validation scenarios they used and why;

2) Assess whether the procedures used are reasonably adequate or whether improved means must be developed;

3) If the existing procedures are generally adequate, then use them to develop a generalized decision procedure for deciding what and how many (based on the Step 1 assessment) scenarios to generate as a function of the features of the particular system under test.

Step 1 was taken. Step 2 revealed generally adequate procedures, so step 3 was followed. It is important when previewing some of the results to note that there were some very strong constraints on the directions concerning validation scenarios. Despite the absence of guidelines or regulations concerning validation scenarios, there is unquestioned agreement within the nuclear power industry as to their purpose and nature, as reflected in the definition developed for this activity:

> *Validation scenarios are realistic dynamic tests of a software (or software and hardware) system which covers only the intended range of application of the software and are designed to sample important subsets of functions, usually for selected situations known to be challenging or problematic, to provide assurance that the system achieves the tested functions with the required accuracy and performance.*

Validation scenarios are used <u>after</u> all the verification and validation testing of the system is completed. Validation scenarios augment the V&V guideline as described in Volume 7 (Miller, et al.) The basic V&V process in Volume 7 is used first to find faults outside the intended range of application. Validation scenarios provide major practical demonstration events of the functional correctness of the developed software. They are not concerned with assessing the multiplicity of code paths or the robustness of the software under stress. These are provided by the detailed V&V guideline packages presented in Volume 5 (cf. Miller, et al, 1993). Rather, the pragmatics of validation scenarios are that they are demonstrations of the system's effectiveness and accuracy in selected realistic situations. Because of this high degree of agreement concerning validation scenarios, this task essentially became one of providing a generalized cohesive and explicit statement of what most workers in the area understand and agree upon implicitly. The challenge was resolving and integrating a wide variety of disparate views about validation scenarios, all of them "correct," and providing a general set of rules which correctly captures each specific insight under a cohesive common framework.

2.1 <u>Importance of Validation Scenarios</u>

Validation scenarios are widely used for the testing of complex software systems, particularly those which involve some safety aspects. They are particularly in wide use within the nuclear utility industry.

Validation scenarios will differ from one application domain to another, but they almost always share the feature that the operational context of the test is one which is known to cause, or reveal, system difficulties within the intended range of application. Within the communications industry, scenarios involving high traffic loads with complex path switching may be used. For missile guidance systems, simulated trajectories over the Arctic circle are often chosen. Within the nuclear utility industry, scenarios frequently involve one of the approximately 20 typical nuclear design basis transients required to be analyzed in the safety analysis reports of commercial nuclear power plants (cf. NUREG-0800, 1981). Putting the system under some kind of important and well-understood operational stress is therefore the key characteristic of validation scenarios.

Validation scenarios can be used informally at any time, but they are formally invoked as a key V&V step typically at the end of implementation V&V, after all the static analyses, the reviews, and the usual structural, domain, and functional dynamic tests have been completed. Therefore, another key feature of these tests is that they are performed on the system as a whole, (even if some of the input data streams are simulated).

An implication of this characteristic is that AI systems will, and should, be treated the same as conventional systems, so far as dynamically generating and executing validation scenarios are concerned. This is because there is nothing about the use of a validation scenario per se which is affected by the programming approach or the internal program structure. Nevertheless, since most of the important behavior of an AI system is contained in the knowledge base component, a novel static analysis approach for generating "quasi-scenarios" from the knowledge base is suggested in Section 6.

2.2 Assessment Approach

The application of selected scenarios for validating AI systems is based on the same reasoning as for validating conventional software. Use of the integrated AI systems tests its overall behavior in much the same way as conventional software keeping in mind the fact that the unique aspect of AI systems (i.e., the knowledge base) has already previously been verified by the knowledge base certification process outlined in earlier tasks of this project. This approach does not consider post-validation (i.e., operational experience) of the developed software. This is considered part of the maintenance component of V&V.

Unlike the efforts to survey and assess software verification methods, the emphasis of this assessment was to contact only those organizations which are actively using production software in the nuclear power industry. In some cases, published reports and documents (A-85-11A, 1986; NSPSAD-8102NP, 1981; NTH-G-6, 1985; NUSCO 140-1, 1984; TVA-TR81-01, 1981; UAI 83-15, 1983; VEP-FRD-19, 1976; VEP-FRD-20, 1977; VEP-FRD-23, 1978; VEP-FRD-24, 1978; VEP-FRD-33, 1979; VEP-FRD-41, 1981; VEP-NFE-2, 1983) submitted to the USNRC were also reviewed if they contained validation scenario information. Most of the software examined in this assessment supports the design and operation of commercial nuclear power plants and has been subject to USNRC review and approval either explicitly or implicitly (USNRC Safety Evaluation Report, Feb. 1983; USNRC Safety Evaluation Report, Nov. 1983).

The key information obtained from each nuclear industry contact or document review can be summarized as answers to the following questions:

1) For specific software, what are the number and types of scenarios which are run before the organization is satisfied that the software has been adequately validated?

2) What is the basis for determining the specific set of validation scenarios?

3) How is the validation process, especially the selection of scenarios, controlled?

The answer to the above three questions provided considerable insight and understanding of the current state-of-the-art and acceptable practice in validation scenario application to nuclear power industry software.

2.3 Assessment Results

A total of 11 nuclear power related organizations were directly contacted and 29 nuclear software documents (see reference section) with validation scenario information were reviewed as part of this assessment. This resulted in obtaining information for 39 software validation cases. The nature of the organizations which provided specific software validation scenario data is graphically presented in Figure 2.3-1 (the details concerning organization, type of scenario used, etc., are given in the next section).

Nuclear utilities and EPRI represent the largest source of validation scenario information with a significant fraction obtained from private nuclear engineering and consulting companies and national laboratories.

Several common underlying themes emerged from the assessment contacts and document reviews. There is no set formula or rule-of-thumb that is used by industry and the government for determining an acceptable number and type of validation scenarios for nuclear software. The closest common thread is that these scenarios should represent the expected range of applications for the software. Thus, for example, if a pressurized water reactor (PWR) nuclear fuel management core physics code is being validated, it would be expected to be validated with fuel loadings of PWR fuel types in the range of uranium-235 enrichment, power levels, burnup, boric acid concentration, and burnable poison that might be seen at an operating PWR. Validation scenarios outside the realm of expected application are not considered or used. This strategy is also applied to the use of options on software. Only option combinations which are physically realistic for the particular software's intended application are included in validation scenarios. It should be noted that previous testing beyond the physically realistic realm of expected applications is included in the basic V&V discussed in Volumes 2, 5 and 7 (Miller, et al). This V&V testing is designed to remove faults outside the expected application range.

Another important conclusion from this assessment is that the selection of validation scenarios is left up to either the code developer, if it is new software, or the end user(s), if it is a revised version of existing software. The subject matter expert usually decides on the validation scenarios although final acceptance of the validation process is left up to a different individual. Since earlier extensive V&V testing which is discussed in Volumes 2, 5, and 7 is performed by a person or organization other than the developer, the selection of validation scenarios by the code developer or subject matter expert is considered to be acceptable for AI systems. Revised versions of already validated software rely on a

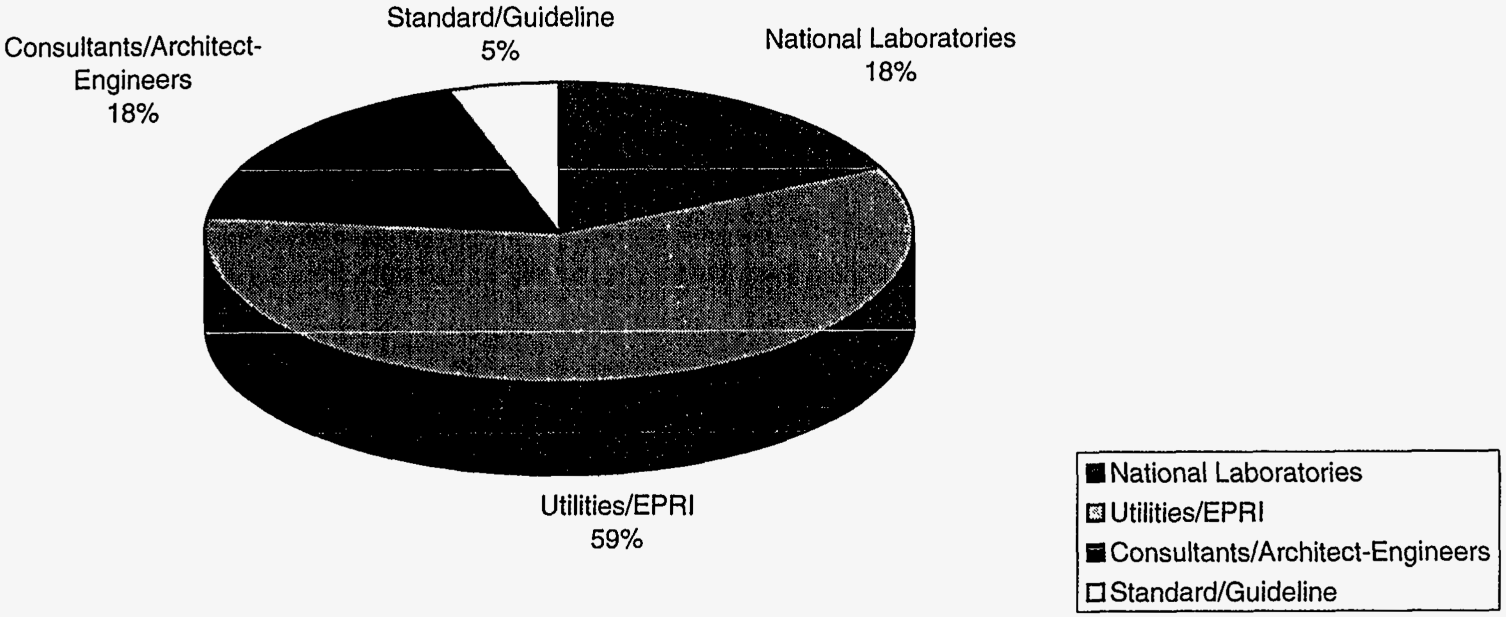


Figure 2.3-1 Assessment Source of Data for Nuclear Industry Software Validation Scenarios

regression suite of validation test cases from the previous version which may be augmented by additional scenarios supplied by the user.

The validation process within reporting organizations appears to be written as a general software quality assurance procedure, usually referencing a general QA or V&V standard such as ASME NQA-2A-1990 Part 2.7 (ASME, 1990). These internal proprietary procedures sometimes will assign organizational responsibility for developing and running validation scenarios but do not provide any specific guidance on the nature or number of validation scenarios to be run.

One trend that was noted during this assessment is that software developed outside the end user's organization (e.g., software developed by EPRI or a consultant) would be delivered to the user with a set of standard validation cases which had been run and could be duplicated on the customer's computer. The customer would typically run additional validation scenarios as a means of ensuring the competency of specific analysts and plant-specific models with the software that may have different features than those provided with the developer's validation scenarios.

The assessment also found that the validation scenarios often encompassed a mix of different types of scenarios, as defined in Section 3.1 (e.g., already validated code cases with plant data and regulatory requirements). Although importance was given to the comparison to actual measurements either at a nuclear power plant or experimental facility, the resulting data was also treated as less than perfect because of uncertainties in initial conditions and instrumentation accuracy. Thus, actual measurements are not treated as the panacea for acceptable software validation in the nuclear industry although they are considered to have the greatest credibility.

3 VALIDATION SCENARIO ASSESSMENT ANALYSIS

An analysis of the results of the validation scenario assessment led to two important conclusions. First, scenarios can be grouped into several distinct categories. Section 3.1 describes the proposed classification scheme for validation scenarios. Second, several factors exist which can affect scenario selection. These factors are described in Section 3.2.

3.1 Types of Scenarios

The nature of validation scenarios can be classified as one of the following types:

1) BASICS -- classical scenarios with "textbook" basic analytical solutions, but which usually are too simple to match the actual situation,

2) CODE -- scenarios which have been executed on identical subject matter software that has already been verified and validated.

3) TESTS -- test results from an instrumented experimental facility,

4) PLANT -- actual measured data from operating nuclear power plants,

5) LICENSING -- scenarios required or recommended by licensing regulations or guidelines, and

6) REGRESSION -- cases that have been run on other previously validated and accepted software with the same function including previous versions of the software being validated. They are composed of any of the above five types and are not truly a separate type.

A compilation of the reported numbers and types of validation scenarios for 39 nuclear power plant-related conventional software validation is presented in Table 3.1-1. The first column gives the name of software programs for which validation scenarios were used; the second column is the judged V&V class of the software; column three gives the name of the organization doing the testing; column four is the judged type of validation scenarios used; and column five is the number of scenarios actually used. This data is graphically presented by number of validation scenarios and by frequency of use in Figure 3.1-1. This figure shows that the validation scenario usage emphasizes Licensing, Plant, and Test types over the Code and Basics types. Regression cases are the least used. However, it should be noted that the relatively low number of software systems using Regression cases reflects the large amount of software in this assessment. When evaluated for modified software, Regression cases are more widely used. Also, some users may consider regression scenarios as another type.

In the context of this project, BASICS scenarios are defined as those validation cases which are derived by the theoretical solution of first principles, equations or relationships which accurately describe physical processes. Since first principles are generally not directly applicable to nuclear power plant modeling, BASICS scenarios are assumed

Table 3.1-1 Conventional software validation scenario assessment results

Software	V&V Class	Organization	Validation Type	# of Scenarios
RETRAN	2	U	LICENSING PLANT	6 7
PDQ07-Discrete	2	U	PLANT	4 FUEL CYCLES[1]
PDQ-07-One Zone	2	U	PLANT CODE	6 FUEL CYCLES 6 FUEL CYCLES
FLAME	2	U	PLANT CODE	5 FUEL CYCLES 5 FUEL CYCLES
RETRAN-MSLB	2	U	LICENSING	2
COBRAIIIC-MIT	2	U	LICENSING	9
RETRAN	2	U	PLANT CODE TEST	5 2 2
DYNODE-P	2	C	TEST	7
RETRAN	2	U	LICENSING PLANT	5 6
RETRAN	2	U	PLANT LICENSING	6 8
LYNX1/2	2	U	LICENSING	4
RETRAN	2	U	LICENSING PLANT	7 3
VIPRE-01	2	U	LICENSING	12
SIMULATE-3	2	U	PLANT LICENSING	2 FUEL CYCLES 13
RETRAN-02	2	U	BASICS LICENSING	4 10
Plant Simulator	2	S	LICENSING	~40
DFCS	2	C	BASICS CODE	~22 ~37
RETRAN-SGTR	2	U	LICENSING	2
SPDS	2	S	LICENSING	29
RELAP5	2	L	TEST	14
TRACBD1	2	L	TEST	19

Table 3.1-1 Conventional software validation scenario assessment results (Cont.)

Software	V&V Class	Organization	Validation Type	# of Scenarios
TRACBD1/MOD1	2	L	TEST PLANT	9 4
TRACPD2	2	L	TEST	7
TRAC-PF1	2	L	TEST	36
RELAP5-MOD1	2	L	TEST	35
TRAC-BD1	2	L	TEST	6
RETRAN	2	U	PLANT CODE LICENSING	9 1 3
RELOAD SAFETY	2	U	LICENSING	16
CORE PHYSICS	2	U	PLANT	3 FUEL CYCLES
RELOAD SAFETY	2	U	LICENSING	16
RELAP5	2	E	TEST	15
PRESTO-B	2	C	PLANT REGRESSION	6 FUEL CYCLES 40
RETRAN02 Mod5	2	U	REGRESSION TEST LICENSING PLANT	17 2 6 7
CALM	2	C	BASICS PLANT	8 2
SPDS	2	C	LICENSING	25
Tech.Spec.Adv.	2	U	BASICS	19
RETRAN-02	2	UR	TEST PLANT LICENSING	15 11 20
VIPRE-01	2	UR	TEST LICENSING CODE	45 14 14
ANSYS	2	C	BASICS TEST REGRESSION	~3800 ~30 ~30

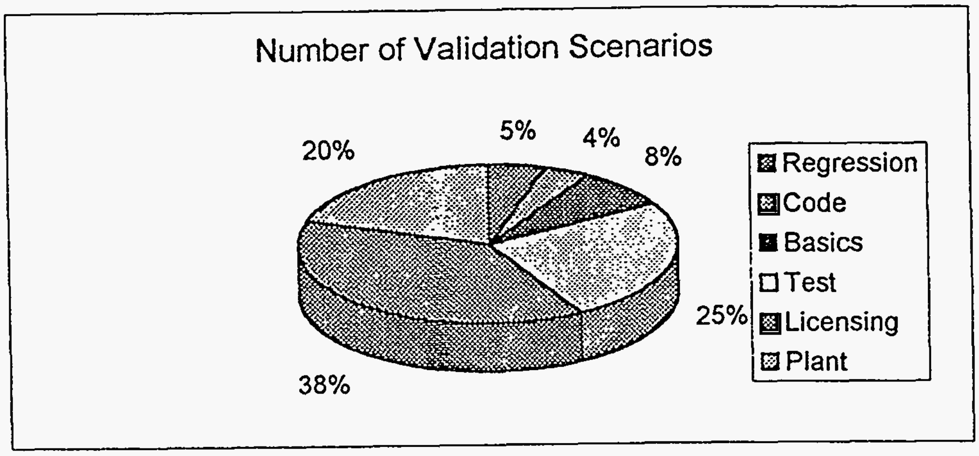
Legend

U	= Nuclear Power Plant Utility
C	= Consultant and/or Computer Software Development Organization
S	= Standard or Guidelines
L	= National Laboratory
UR	= Utility Research Organization

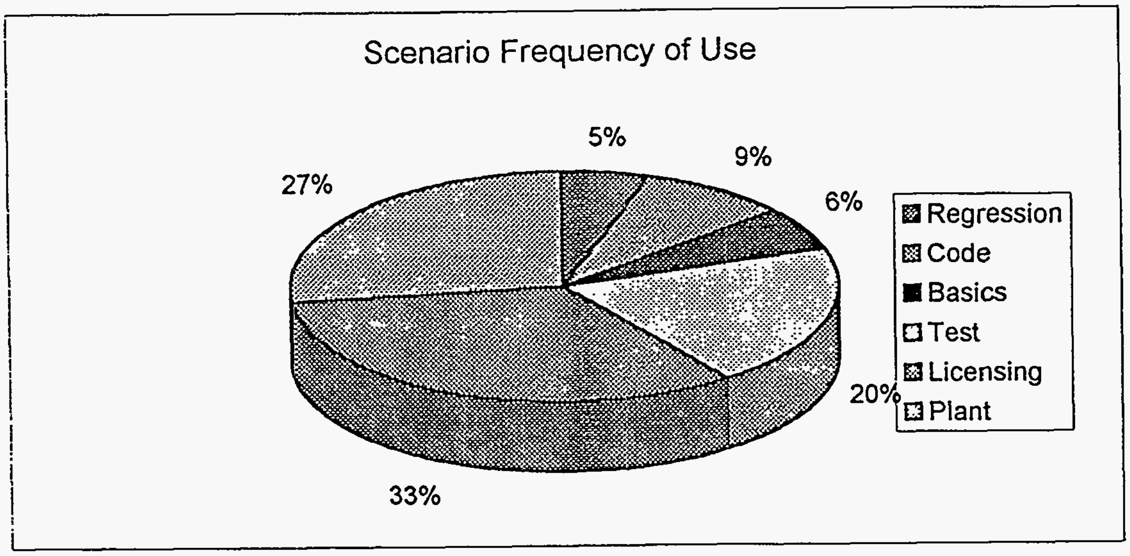
[1] A fuel Cycle Scenario represents measured plant data, usually taken monthly, from incore probes of nuclear power distribution within the fuel assemblies as well as other reactor physics parameters such as control rod reactivity worth, critical soluble boron concentration (for PWRs) and power reactivity coefficients.

Figure 3.1-1
Validation Scenario Survey Results: Scenario Type Distribution

	Regression	Code	Basics	Test	Licensing	Plant
Number of Validation Scenarios	5%	4%	8%	25%	38%	20%



	Regression	Code	Basics	Test	Licensing	Plant
Scenario Frequency of Use	5%	9%	6%	20%	33%	27%



to be those scenarios which make approximations or simplifications to either the real plant situation or first principles so as to predict specific nuclear power plant behavior.

It has been the experience of the authors of this report, in conjunction with an assessment of nuclear software literature, that most nuclear power related software is not amenable to analytical solution validation. It should also be noted that, of the 39 specific examples of nuclear software validation presented in Table 3.1-1, only five systems cite the use of BASICS scenarios. The greatest use of BASICS scenarios is in the case of ANSYS which is a very large general purpose finite element stress analysis computer code. Each ANSYS application can be simplified to correspond to classic textbook structural analysis solutions. Most nuclear software cannot be so simplified for this type of validation. In fact, the reason for creating much of the software is that nuclear engineering theory is too complex for straight forwardsolution. Frequently, parameters must be solved for in space, time and energy dimensions simultaneously. Alternatively, phenomena are modeled which are not well understood by theory requiring considerable empiricism or inherent conservatism.

Software CODE scenarios represent the use of results from scenarios executed on another software system which have been verified and validated and are generally accepted in the nuclear industry as a tool for performing the same functions as the software being validated. It should be noted that the software system which has been verified and validated may not necessarily have received the nature and extent of V&V that is presented in this project. In addition, the acceptance of the system by the nuclear industry is not meant to imply that it has been officially accepted by the USNRC, but rather that subject matter experts in the nuclear industry generally accept the use of this particular software for performing certain analyses or functions related to nuclear power plants.

Usually, the already verified and validated identical subject matter software system utilizes a different methodology to calculate the results. Thus, some differences between the two systems' scenario results are to be expected, but should be explainable in terms of the difference in solution techniques. CODE scenarios are important in that they demonstrate a system's capabilities in terms of another known and accepted software system. They are limited by the fact that the software system being compared to may still have some errors or other shortcomings and is, in itself, only a means of modelling physical phenomena.

TEST scenarios are most often based on facilities which are designed primarily to understand phenomena, but not for software V&V. These facilities have extensive calibrated instrumentation, data acquisition systems, and other design features intended to measure a wide range of parameters of interest in establishing validation scenarios. One disadvantage of TEST scenarios is that instrumentation failure, inaccuracy or drift can, and often does, occur. This could degrade the quality of the results. Another problem with TEST scenarios is the fact that these costly facilities are almost always constructed on a reduced scale with other design compromises when compared to actual operating nuclear power plants. Thus, even a good comparison between the software predictions and experimental facilities leads to questions as to the adequacy of the software in modeling full scale nuclear power plant behavior.

PLANT scenarios are generally viewed as one of the most important means of validating software since the data does not represent any compromise from the actual expected nuclear plant's behavior. Certain tests and measurements are routinely performed or required for operating nuclear power plants. These include a wide range of startup tests involving pump flow, turbine-generator load, core power distribution, and reactivity coefficients. During operation, the core power distribution is periodically monitored as well as key thermal-hydraulic parameters which are inputs to technical specifications and protection systems. This data, with its proper qualification, is a valuable source of validation scenarios, especially for transient behavior. The disadvantages of PLANT scenarios include the potential unavailability of useful data for some software. A particular type of software may involve a technical area that is not directly measured by instrumentation. For example, nuclear fuel rod failure can be analyzed and predicted by some computer codes, but there is no direct way to measure if a specific fuel rod has failed and any details of its failure mechanism while it is operating. Only costly post-mortem laboratory analyses can provide that type of data. Another negative aspect of plant data is the very nature of the data. In real world operating situations, instruments fail, lose accuracy and precision, and drift. Data collection may have gaps or unexplained oscillations and the exact status of equipment may not always be known. Documentation of plant data can vary greatly in both quantity and quality. Also, operator actions may occur which influence the usefulness of the data. All these factors need to be accounted for in order to use plant scenarios for useful software validation. An additional limitation is that plant data is usually only available for a limited range of the situations (i.e., normal operating conditions) needed to validate the software.

LICENSING scenarios are only applicable to validation of software which involves disciplines directly under the auspices of specific USNRC regulations or guidelines such as the Standard Review Plan (NUREG-0800) or the Code of Federal Regulations. In such cases, regulations specify scenarios that need to be analyzed. A prudent validation would include some of these regulatory scenarios to demonstrate code capability and the user's ability to model the plant and properly use the software. Since licensing scenarios are conservative and not realistic, the only point of comparison for the software results would be the results of other software which has been accepted by the USNRC (e.g., NSSS and fuel vendor software for reload safety analyses). Thus LICENSING validation scenarios are important for that subset of software which is within the purview of regulations that delineate scenarios.

The last type, REGRESSION, is not really a different substantive type at all, since it is composed of the preceding five types of scenarios. However, REGRESSION scenarios are spoken of, in the reference sources, as if they were a different type because they are employed differently than the five types. That is, REGRESSION cases are specially selected cases intended to reveal whether any <u>modifications</u> to the software system changed its functional operation on non-modified software system aspects, compared to what the operation used to be. Users often have a special suite of regression cases with which they are highly familiar and will use any time there is a change to the software (and even hardware) system.

3.2 Factors Affecting Scenario Selection

As the aforementioned analysis indicates, each of the six types of validation scenarios has its merits and drawbacks or limitations. Factors that will affect the type (or mix) of validation scenarios include:

1) Applicability to specific software,
2) Availability of plant or experimental scenario data,
3) Software V&V Class (as defined in this project; see Volume 5),
4) Presence of previous code version validation scenarios, and
5) Existence of identical subject matter validated software.

The first factor of specific software <u>applicability</u> affects the relevance of LICENSING scenarios since, if the software does not fall under the auspices of any specific regulatory guidance that sets scenarios, this type is not applicable. The second factor of data availability will dictate if plant or experimental facility scenarios can be used for validation. Software complexity and degree of required integrity determine the V&V class, and this third factor should be used as a guide in selecting the range of validation scenarios. A code that deals with one parameter (e.g., performance implied from pump vibration) would require fewer scenarios than a complex code with a wider range of application (e.g., pump and connected piping system transient behavior to postulated accidents). If other already validated and accepted software exists which analyzes the same subject matter, comparison to these previous code validation scenarios represents a valid component of the validation program. Finally, the fifth factor specifically addresses the case where the software being validated is a modification or revision of existing validated software.

4 METHODOLOGY FOR SELECTING VALIDATION SCENARIOS

Analysis of the assessment results led to a classification of scenario types and identification of the factors affecting scenario selection. Using this as a foundation, a methodology for selecting validation scenarios has been developed. The following sections discuss the basis for the methodology and the guidelines for actually selecting validation scenarios. The guidelines discussed in this section were derived by a combination of an assessment of the information gathered for this task and the technical judgment of the authors. While considerable information and a high degree of confidence exist regarding V&V Class 2 and 3 system validation scenarios, there is less information and confidence on the guidelines for V&V Class 1 software.

4.1 Underlying Considerations

Validation scenarios are one of the final means by which software is shown to implement the approved design. This demonstration is subjective to the extent that there is no quantifiable way to assure the adequacy of a set of validation scenarios. A good set of scenarios must consider the range of application for the software and a comparison to some type of irrefutable data whether it is in the form of plant or experimental facility measurements, other accepted software results, or basic first principles. In those instances where the software institutes operating procedures of some kind which are not so tightly linked to complex physical plant computations, then the validation preference is usually a written guideline or procedure. However, good engineering judgement is always required to determine that the software recommendations or actions are reasonable.

For validation scenario selection, the software must first be classified so that the relevance of each type of scenario which is discussed in Section 3 can be evaluated. The overriding consideration is applicability. If the nature of the software is such that no basic first principles solutions apply, then this type of scenario cannot be used. The same rationale will determine whether a software's validation can utilize experimental, plant, or licensing data. Regression tests are only of value if the software represents a modified version of an already validated code or if already validated software with the same function exists with its own scenarios.

4.1.1 New Software

After classifying the software to ascertain which types of scenarios can be used, a hierarchy of scenario types is applied. Within certain boundaries and data limitations, the following order of importance to validation scenarios is delineated for NEW software (in descending order of importance):

1) PLANT
2) TEST
3) BASICS
4) CODE
5) LICENSING

The basis for selecting an order of importance or hierarchy of validation scenario types is that of how closely a scenario type represents the real nuclear plant. Since the system is intended to simulate some aspect of the plant's actual behavior, validation with "real world" information directly measured at the plant would substantiate a system's modelling capability. Experimental facilities which are designed to perform investigations into specific aspects of some detail of plant behavior are scaled and instrumented so as to provide large quantities of information on phenomena that were not well understood. This test data represents a "step back" from actual plant measurements, but does constitute a source of detailed information on phenomena that cannot be examined at operating plants or usually represented by first principles. Therefore, test data is another validation scenario type which represents a source of real world measurements. Thus, the principle used in determining the order of importance for validation scenario types can be summarized by the following question: "How closely does the validation scenario represent the real expected behavior of a nuclear power plant?"

The desirability of PLANT data for scenarios is directly related to the quality of the data. This includes detailed knowledge of the plant configuration and design, initial conditions, instrumentation accuracy, location, and precision, and operator actions. Without highly qualified plant data, this type of validation scenario is still useful, but subject to engineering judgement and limited applicability.

The same range and level of detail of information is needed for experimental facility TEST scenarios. If the test was not properly instrumented or instruments failed or exhibited unexplained erratic behavior this scenario type is still useful, but subject to engineering judgement and limited applicability.

In the context of this task, BASICS scenarios are defined as those validation cases which are derived by the theoretical solution of first principles equations or relationships which accurately describe physical processes. In ideal circumstances, first principles would be directly applied to a process that is modeled by computer software, and the results could be directly compared as part of the evaluation of the results of this validation scenario.

An ideal example of BASICS scenarios is the calculation of mass of water added to a vessel when the total mass introduced at a water-tight cold pipe emptying into the vessel is known. Based on conservation of mass, a first principle, the mass added to the pipe is equal to that entering the connecting pipe and can be used to validate computer software calculation of that parameter. Unfortunately, in the real world, ideal situations rarely exist. Scenarios modeled by nuclear power industry software usually require assumptions, simplifications, extrapolations, and other means to obtain the exactness of the theoretical analyses. Thus, in reality, the pipe connected to the vessel may be connected to other pipe or may, have a leak, and the vessel may contain high temperature and pressure steam which cause a three-dimensional countercurrent flow in the pipe suspending or preventing some of the water flow from entering the vessel. In this more accurate situation, first principles would need to be augmented by additional correlations and simplifications or possibly replaced by experimental data.

For this discussion BASICS scenarios will therefore be interpreted as those cases which are based on first principles and are capable of validating a section of the software, but not its completely integrated function. If software could be completely modeled by first principles, without simplification or modification, then BASICS scenarios would be more important than they are presently ranked.

Finally, BASICS first principle validation scenarios are important in looking at small parts of the software rather than the entire code, but they can also be useful in validating overall trends without considering the accuracy of specific numerical parameter values.

CODE scenarios offer an ability to validate system functions or capabilities that cannot be confirmed with PLANT, TEST or BASICS scenarios. The strength of this scenario type is based on the perceived fidelity of the other software system which it is being compared to in terms of its own V&V and acceptability in the nuclear industry. The weakness of CODE scenarios is the fact that they represent a software-to-software comparison rather than a comparison to measured real world data or well known first principles solutions.

The value of LICENSING scenarios must be taken within the context of the regulatory requirements for these cases.

4.1.2 Modified Software

To ascertain if software is modified and not new, engineering judgment is required in evaluating the nature and extent of the changes in capability of the software. For MODIFIED software which does not represent a major change to the code, the following order of importance for validation scenario types is applicable:

1) REGRESSION
2) PLANT
3) TEST
4) BASICS
5) CODE
6) LICENSING

REGRESSION scenarios are valuable as long as the modified new software version does not represent a large deviation from the previous version's capabilities. Major modifications will require further validation beyond the last version's set of scenarios. Another interpretation of Regression scenarios are those from already validated and accepted software which perform the same function, but use different methods.

For this list, it is presumed that the regression set of scenarios includes a previously acceptable mix of PLANT, TEST, BASICS, CODE, and LICENSING scenarios as they apply to the application. Following application of regression scenarios, the order of priority for the main types of scenarios is the same as for new software (see Section 4.1.1). Thus, the purpose of differentiating between new and modified software is in elevating the importance of REGRESSION scenarios.

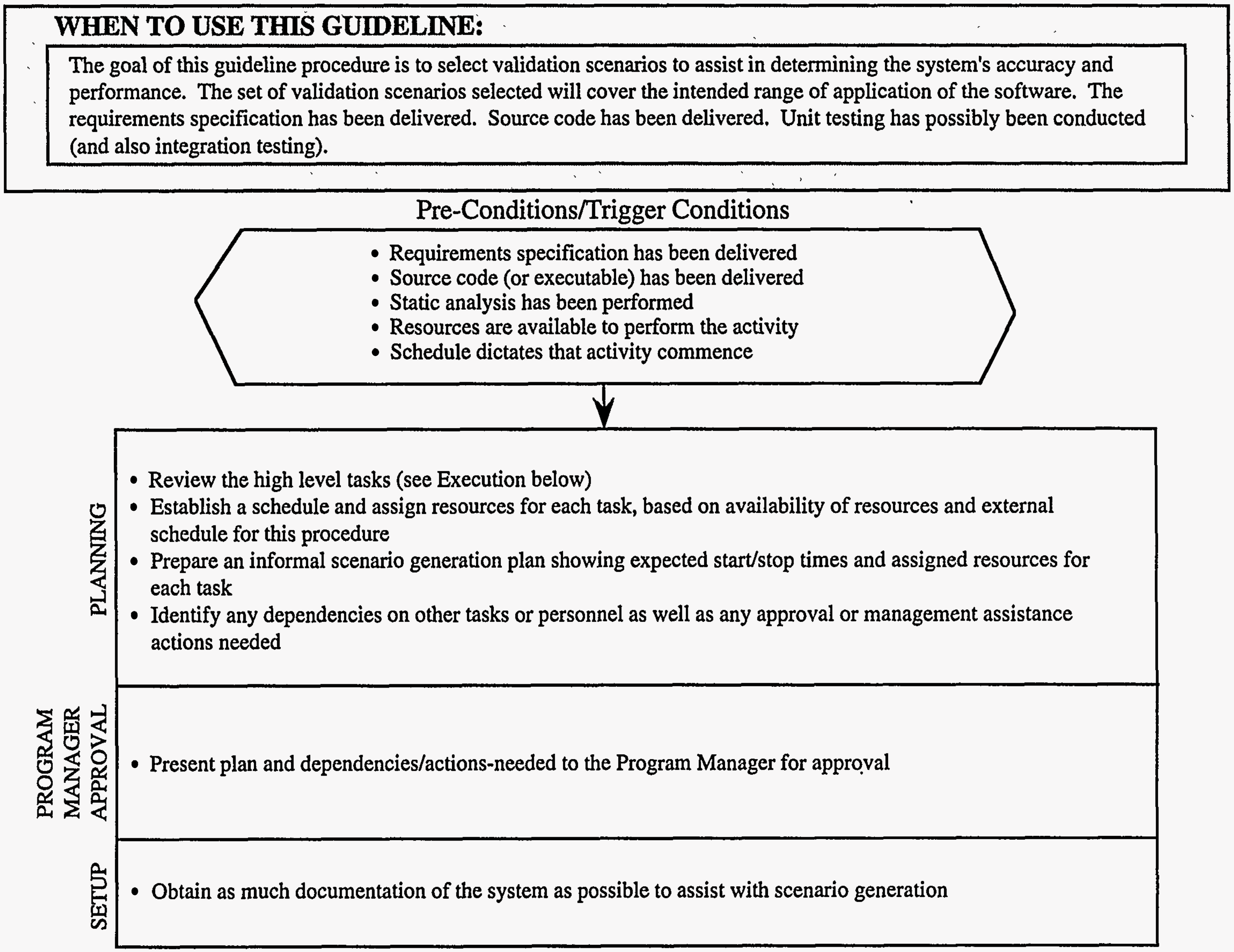
4.2 Selection Strategy

The underlying philosophy for selecting validation scenarios, within the constraints of well qualified data, is to place the highest importance on real world measurements either at operating nuclear power plants or experimental test facilities. Since the ultimate application of the software is to support the operation of nuclear plants, validation of their behavior is paramount. The desirability for these scenario types, however, must be tempered with the knowledge that such data may not exist or be sufficiently qualified. Also, some software applications are not based on best estimate modeling, but rather on licensing (i.e., conservative) bases. In this case, licensing scenarios constitute a good source of validation scenarios. BASIC first principles scenarios have a place in the mix of validation scenarios in that they can be used to prove individual parts of the system and validate trends. CODE scenario comparison can be a useful component of validation, but is dependent on the reputation and pedigree of the system being compared to for evaluation. Finally, REGRESSION cases, which actually consist of some mix of the previously discussed five types, are of the highest importance for modified versions of existing software because they are a validation suite whose results are already understood. They play a smaller role for completely new software where the validation suite is new.

4.3 Validation Scenario Guidelines

The guidelines for selecting validation scenarios are based on the key concepts described in Section 3. These concepts are type of scenario and factors affecting scenario selection. This information was used to develop a guideline procedure, in the same style and format as those provided for validation scenarios. The guideline procedure is shown in Figure 4.3-1. Following this procedure will result in knowing what types of validation scenarios to use and how many scenarios of each type. The actual selection of the specific test cases; however, requires the assistance of an engineer or similarly experienced and trained professional, to pick the specific test-case parameters appropriate for the system context. The nature of the considerations underlying such choices are discussed in the next section.

Figure 4.3-1 SELECTING VALIDATION SCENARIOS



WHEN TO USE THIS GUIDELINE:

The goal of this guideline procedure is to select validation scenarios to assist in determining the system's accuracy and performance. The set of validation scenarios selected will cover the intended range of application of the software. The requirements specification has been delivered. Source code has been delivered. Unit testing has possibly been conducted (and also integration testing).

Pre-Conditions/Trigger Conditions

- Requirements specification has been delivered
- Source code (or executable) has been delivered
- Static analysis has been performed
- Resources are available to perform the activity
- Schedule dictates that activity commence

PLANNING

- Review the high level tasks (see Execution below)
- Establish a schedule and assign resources for each task, based on availability of resources and external schedule for this procedure
- Prepare an informal scenario generation plan showing expected start/stop times and assigned resources for each task
- Identify any dependencies on other tasks or personnel as well as any approval or management assistance actions needed

PROGRAM MANAGER APPROVAL

- Present plan and dependencies/actions-needed to the Program Manager for approval

SETUP

- Obtain as much documentation of the system as possible to assist with scenario generation

- 1) Classify the software to be validated using scenarios:

  - Examine the following five categories of scenarios and consider the applicability of each to the software (e.g., if the nature of the software is such that no basic first principles solutions apply, then this type of scenario would not be selected). Review the list below to determine which types could be used:

    - BASICS scenarios → classical scenarios with "textbook" basic first principles solutions
    - TESTS scenarios → scenarios utilizing experimental facility instrumented test results
    - REGRESSION scenarios → a mix of the other five types of scenarios that have been used on previously validated and accepted software which performs the same function and should be executed again on the modified software to ensure that previously working functionality has not been corrupted
    - CODE scenarios → scenarios using cases from identical subject matter software which has been fully verified and validated
    - PLANT scenarios → scenarios using actual measured qualified[1] data from operating nuclear power plants
    - LICENSING scenarios → scenarios required by licensing regulations and guidelines

  - List separately all scenario categories applicable to the software and proceed to Step 2.

- 2) Software falls into two categories: 1) all other software referred to as NEW (includes mixture of new software and significant modification to old); 2) software that represents a minor modification of an earlier already validated version(called MODIFIED).

  If the software is NEW, use (in descending order of importance):

  1. PLANT scenarios
  2. TEST scenarios
  3. BASICS scenarios
  4. CODE scenarios
  5. LICENSING scenarios

  If the software is MODIFIED, use (in descending order of importance):

  1. Appropriate REGRESSION scenarios
  2.. PLANT scenarios
  3. TEST scenarios
  4. BASICS scenarios
  5. CODE scenarios
  6. LICENSING scenarios

[1] Qualified data implies that plant initial conditions and equipment status as fully known, that all operator actions affecting the data are documented, and that the location, accuracy, and precision of data measuring instrumentation is also known.

EXECUTION (cont.)

- 3) Determine the degree to which the software should be exercised by scenarios:

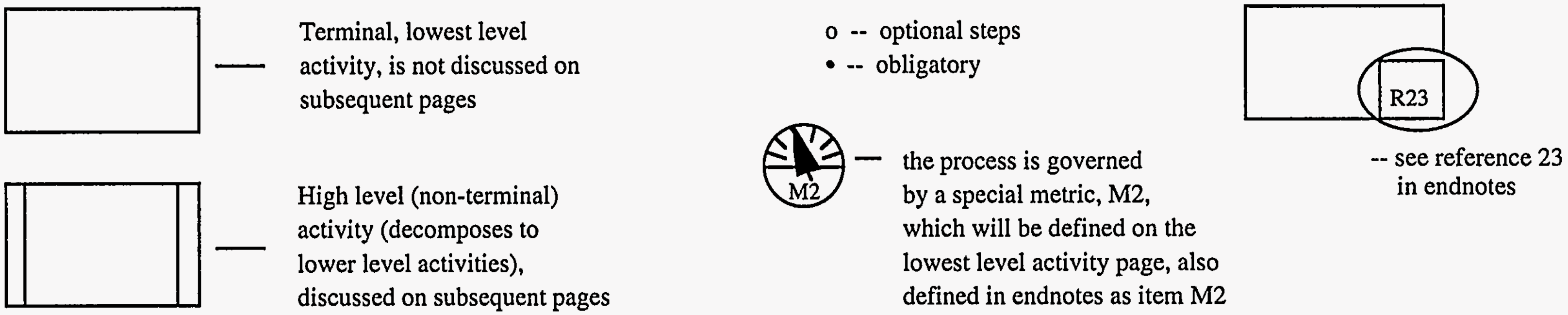
   1. Determine the software V&V Class (Class 3, 2, or 1, as defined in Volume 5, see Figure 5.3-1).
   2. Determine whether the software is NEW or MODIFIED.
   3. The least the software should be exercised by scenarios is to pick a few which cover much of the functionality of the system, including any aspects which were modified.
   4. MODIFIED software for the least stringent V&V Class, Class 3, will need the least number of scenarios.
   5. As the V&V class increases in stringency, additional scenarios should be added to insure increased cover of functionality and to more widely sample the input space. For Class 1 software, sufficient scenarios should be chosen to explore those aspects which could lead to hazards or catastrophic failure or which are otherwise very challenging or problematic.
   6. NEW software should have more scenarios than MODIFIED.
   7. Remember that the purpose of validation scenarios is to provide assurance to the users, customer, and regulators that the overall system is performing to specification. The purpose is not to severely stress the program or greatly sample possible inputs. Those things will have been accomplished by the previous validation tests.
   8. Table 3.1-1 can be consulted to get an idea of the number of scenarios others have used. This table shows that from 2 to 3860 scenarios have been used with the majority of cases using about 10 to 70 scenarios.

- 4) Adhere to the following rules to determine which types of scenarios to include:

   1. The prioritized sets of categories from Step 2 (for NEW and MODIFIED software) are to provide the primary guidance.
   2. For NEW software, try to satisfy the needed number of scenarios from the set of available PLANT scenarios.
   3. In general, only select scenarios from the next most important category whenever (a) the available scenarios from the most important category have been exhausted, (b) the scenarios are not covering the range of intended applications of the software system, or (c) these scenarios are somehow inappropriate.
   4. Any applicable LICENSING scenarios must be included whether the software is NEW or MODIFIED.
   5. For MODIFIED software it is essential to use some appropriate REGRESSION scenarios. However, do not attempt to obtain all the scenarios from this category. Rather, use a sufficient number to give confidence that the modifications have not altered prior unmodified function, and then set PLANT scenarios as the most important category from which to select the remaining scenarios

EXECUTION (cont.)	• 5) In picking or generating specific scenarios to be used, follow these guidelines: 1. Try to identify relevant regulatory or guideline documents which cover the functionality provided by the system under test. 2. If such documents are not available try to identify less formal guides or local (plant) documents which are relevant. 3. Within the above documents, locate references to standard, customary, or recommended operating scenarios or situations that are either mandated to be considered, or at least characterized as important and challenging. 4. Identify other challenging operating situations from tests of other systems, from other organizations, etc. 5. If no scenarios are discovered, or if the number found is insufficient, then generate scenarios according to step •6). 6. From all the discovered candidate scenario situations select a set which best covers the range of intended application. • 6) To generate validation scenarios, follow the steps below: 1. Review the range of operating conditions the system is intended to operate in; also review the concept of operations. 2. From experience, select a situation context which is complicated, tricky, or involves considerable demands on systems performing in those situations. Ensure that this situation would be understandable and similarly evaluated by your respected peers. 3. For this situation, devise an initial triggering event or cause that will initiate the situation chosen in 2. 4. From engineering analyses, work out a set of reasonable following states that could occur for the external situation given appropriate responses of the system under test. These can include hazards and risky conditions. Determine an appropriate concluding end-state that the system should achieve. 5. Steps 2-4 constitute generation of a single scenario. Repeat as necessary.
ANALYSIS	• The type of validation scenarios will be determined M1 • The number of validation scenarios per type will be determined M2
ACCEPT/ REJECT CRITERIA	• ACCEPT if: Reasonably close agreement with the data source • CONDITIONALLY ACCEPT if: Agreement with the data source requiring some explanations, extrapolations, or interpolations to explain any significant difference. • REJECT if: Results are significantly different from data source and cannot be adequately explained

REPORTING	• Prepare the Validation Scenario Plan
PROGRAM MANAGER REVIEW	• Present the Validation Scenario Plan to the Program Manager
OUTPUT/ COMPLETION STATUS	if ACCEPT → continue to next activity if CONDITIONALLY ACCEPT → correct deficiencies and repeat this activity if REJECT → correct deficiencies, hold review with program manager to determine course of action



— Terminal, lowest level activity, is not discussed on subsequent pages

— High level (non-terminal) activity (decomposes to lower level activities), discussed on subsequent pages

o -- optional steps
• -- obligatory

— the process is governed by a special metric, M2, which will be defined on the lowest level activity page, also defined in endnotes as item M2

-- see reference 23 in endnotes

METRICS & REFERENCES

M1 - The scenario type to be used in the validation

M2 - Number of validation scenarios for each of the types as identified in M1

R1 - Miller, Lance, Jane Hayes, Steve Mirsky. *"Task 7: Guidelines for the Verification and Validation of Artificial Intelligence Software Systems."* Report prepared for the United States Nuclear Regulatory Commission and the Electric Power Research Institute, May 1993.

5 SYSTEM A AND B VALIDATION SCENARIOS

Using the methodology described in Section 4.0, scenarios were selected for both of the experts system used throughout this project to test the recommendations. These systems are identified as Systems A and B (see Section 1.1). A discussion of this implementation of the methodology follows.

5.1 System A Validation Scenarios

In determining validation scenarios for System A, the guideline procedure of Section 4.3 was followed. Based on those results, a minimum of 12 to 20 validation scenarios should be applied in a mix of qualified plant and licensing[1] cases with emphasis on the plant data cases. The selection of licensing scenarios is somewhat subjective since there are no specific regulatory guidelines on emergency operating procedure tracking systems, but there are guidelines on emergency operating procedures and the nature of abnormal plant events which should be covered by these procedures. The next step in this process was to identify the specific scenarios for validation of System A from the available data on plant events and licensing guidance. The availability of fully documented qualified plant tests applicable to System A is small.

For System A PLANT validation scenarios, the obvious source of data for a utility would be its own nuclear power plant's data files for startup tests and unusual occurrences. Each nuclear power plant performs extensive tests during its initial startup for commercial operation to ensure that all important systems and components operate in accordance with their intended design. In addition, during operation, most plants have experienced abnormal conditions which must be reported to the USNRC in accordance with Title 10 Part 50.73 of the Code of Federal Regulations in the form of a Licensee Event Report (LER). These LERs are analyzed by the USNRC, and if significant from a safety standpoint, the USNRC performs further detailed investigations and evaluations.

Openly available sources of plant data were investigated by examining key USNRC and EPRI documents. The USNRC's Office for Analysis and Evaluation of Operational Data (AEOD) is responsible for analyzing all nuclear events and evaluating their significance for nuclear safety. AEOD issues an annual report (NUREG-1272) which summarizes events at operating nuclear plants in the United States. and highlights those events which were deemed significant.

Using Incident Investigation Teams (IITs) and Augmented Inspection Teams (AITs), the USNRC evaluates some events in great detail to determine their causes and broad industry safety implications. Although the USNRC receives a large number of notifications and LERs from nuclear power plant operating utilities, only a very small fraction warrant IIT or AIT attention. For example, in 1991, AEOD received about 1900 LERs which resulted in one IIT and 15 AITs. The level of detail and importance of an AIT or IIT make documents issued by these USNRC teams a prime source of plant data for validation scenarios. The USNRC Office of Nuclear Reactor Regulation (NRR) maintains a database of all AIT reports since 1985.

[1] According to Step 4.9 of the procedure, applicable LICENSING scenarios are recommended for inclusion in all selections in order to ensure that the specific regulatory concerns exemplified by these scenarios are covered. Step 4.3 specifies the conditions for including scenarios of other types than PLANT. Thus, the selection of this scenario mix was based on affording greater credibility to the validation of System A.

In addition to NUREG-1272, AEOD has issued a document that investigates the importance of actual plant events as a precursor to severe accidents (NUREG/CR-4674). This document uses probabilistic risk assessment techniques to evaluate the importance of reported plant events.

Information from NUREG-127 and, NUREG/CR-4674 and a listing provided by USNRC-NRR of AIT reports was analyzed to select a subset of reports which could provide real plant data for System A validation scenarios. The following PLANT scenarios and their reference documents resulted from this analysis for System A.

1) Peach Bottom Unit 2 Cycle 2 Transient and Stability Tests, EPRI NP-564, June, 1978.

2) Excessive Cooldown Rate Event at LaSalle Unit 1, USNRC AEOD Technical Review No. T417, August, 1984.

3) Feedwater Oscillations Resulting in Reactor Trip at Dresden Unit 3, USNRC AIT Report No. 8729, October 16, 1987.

4) Loss of Offsite Power at Vermont Yankee, USNRC AIT Report No. 9113, June 6, 1991.

5) Automatic Depressurization System (ADS) - Reactor Core Isolation Cooling (RCIC) System Interaction Events at River Bend Unit 1, USNRC AEOD Technical review No. T610, December, 1986.

6) Brown's Ferry Unit 3 Partial Failure to Scram Event, USNRC AEOD Case Study Report No. C001, July, 1980.

These six were considered to be the best PLANT validation scenarios applicable to System A. A minimum of six additional scenarios are suggested according to the guideline procedure which recommends they be sampled first from TEST scenarios, then BASICS, then CODE, and finally LICENSING. No validation scenarios were judged appropriate from the first three of these categories, so the remaining scenarios were selected from the last, LICENSING, category.

For LICENSING cases applicable to System A, two USNRC documents were consulted. Regulatory Guide (R.G.) 1.70 describes the format and content of Safety Analysis Reports (SARs) for nuclear power plants which includes a section on accident analysis (Chapter 15 of R.G. 1.70) (cf. USNRC 1982). In addition, NUREG-0800 comprises the USNRC Standard Review Plan (SRP) which includes sections on specific transients in Chapter 15 of the SAR. Both of these regulatory documents discuss the same specific accidents or transients, but from different perspectives and level of detail. The events can be categorized into the following different basic initiating phenomena:

1) Increase in heat removal by the secondary system;

2) Decrease in heat removal by the secondary system;

3) Decrease in reactor coolant system flow rate;

4) Reactivity and power distribution anomalies;

5) Increase in reactor coolant inventory;

6) Decrease in reactor coolant inventory;

7) Radioactive release from a subsystem or component; and

8) Anticipated transients without Scram (ATWS).

Each of these categories actually encompass a number of postulated transient events which are described in greater detail in NUREG-0800. Since each category would test different features of System A, a sample of at least one from each group was selected for the LICENSING validation scenarios. The selected scenarios are:

1) Steam pressure regulator malfunction or failure that results in increasing steam flow.

2) Loss of normal feedwater flow.

3) Coincident loss of onsite and offsite alternating current power to the plant.

4) Malfunction of the recirculation loop controller that results in decreasing flow rate.

5) Control rod drop.

6) Inadvertent closure of main steam isolation valves (MSIVs).

7) Main steam line break.

8) MSIV closure ATWS.

9) Loss of feedwater ATWS.

10) Main feedwater line break.

With the above delineated plant events, a total of 16 validation scenarios have been selected for System A.

5.2 System B Validation Scenarios

In determining validation scenarios for System B, the Section 4.3 guideline procedure was applied. Based on the results, a minimum of 12 to 20 validation scenarios should be applied. A mix of qualified plant, basics, and licensing cases with emphasis on available plant data cases was selected because of the availability of Plant cases, applicability of Licensing cases, and appropriateness of Basics cases to System B. The selection of licensing scenarios is somewhat subjective since there is no specific regulatory guidelines on monitoring critical safety functions[2], but there are guidelines on associated systems such as the safety parameter display system (SPDS) and the nature of abnormal plant events which could be covered by these procedures. The next step in this process was to identify the specific scenarios for validation of System B from the available data on plant events, other software cases, basic scenarios and licensing guidance.

In selecting PLANT scenarios for System B, the same process described in Section 5.1 for System A was used.. This involved an analysis of openly available documents from the USNRC and EPRI. The only difference was that, whereas for System A, BWR plant events were evaluated, in the case of System B, only PWR events were examined for applicability. The selected PLANT scenarios for System B are:

1) Loss of Integrated Control System (ICS) Power and Overcooling Transient at Rancho Seco on December 26, 1985, NUREG-1195, February, 1986.

2) Steam Generator Tube Rupture Event at Ginna on January 25, 1982, NUREG-0909, April, 1982.

3) Power Operated Relief Valve (PORV) Actuation Resulting in Safety Injection (SI) Actuation at Calvert Cliffs, USNRC AEOD Engineering Evaluation Report No. E320, September, 1983.

4) Loss of Main and Auxiliary Feedwater Event at the Davis-Besse Plant on June 9, 1985, NUREG-1154, July, 1985.

5) McGuire Overpressurization Event of August 27, 1981, USNRC AEOD Engineering Evaluation Report No. E248, November, 1982.

6) Main Feedwater (MFW) Pump Suction Line Rupture at Surry Unit 2 on December 9, 1986, USNRC AIT Report No. 8642, February 10, 1987.

7) Loss of Offsite Power and Reactor Trip at Zion Unit 2 on March 21, 1991, USNRC AIT Report No. 91006, April 17, 1991.

8) Steam Generator Boiled Dry Event at Indian Point Unit 2 on January 3, 1988, USNRC AIT Report No. 8803, March 14, 1988.

Suggested BASICS scenarios for system B involve introducing a malfunction of a single plant component or deviation in a single plant parameter that is well understood to affect the status of one of the critical safety functions which is the important output of this system. Variations of the BASICS scenarios listed below could be used, if the variation represented a significantly different validation test of the software. The six CSFs are: reactivity control, reactor coolant system (RCS) inventory control, RCS pressure control, RCS transport control, RCS integrity control, and

[2] It should be noted, however, that GDC13 and RG 1.97 provide guidance on safety parameters which are input to the status for the critical safety functions.

RCS heat sink control. The six BASICS scenarios, all initiated from a steady state 100% power operation state for a four loop PWR, selected for these CSFs are:

1) Five Control Rods Stuck Out and Failure of Boron Injection.

2) Pressurizer Level at Technical Specification Maximum Level and RCS Charging System Operating without Letdown or Shutoff Capability.

3) Stuck Open PORV Resulting in a Cooldown rate greater than the Cooldown Pressure Limits.

4) Plant trips from full power, three loops in natural circulation, the third loop hot-to-cold leg temperature difference equals 100 °F.

5) RCS Pressure rises above Technical Specification Limit.

6) Steam Generator A level drops below minimum natural circulation level after plant trip with the remaining three steam generators at above minimum level.

The same USNRC documents applicable to System A (see Section 5.1) are applicable to System B, and the same eight initiating phenomena apply. However, given that System B is intended for PWRs and evaluation of critical safety functions, a different set of scenarios are appropriate. The selected scenarios are:

1) Equipment failure that results in increasing main feedwater flow rate.

2) Loss of normal feedwater flow.

3) Coincident loss of onsite and offsite a.c. power to the plant.

4) Inadvertent opening of a steam generator safety valve.

5) Control rod ejection.

6) Reactor coolant pump shaft break (locked rotor).

7) Main steam line break.

8) Uncontrolled control rod assembly withdrawal at power.

9) Inadvertent operation of high pressure safety injection system during power operation.

10) Steam generator tube rupture.

11) Loss of coolant accident (primary coolant pipe guillotine break) at power.

12) Loss of load ATWS.

A total of 26 validation scenarios have been recommended for System B which meets the suggested range of 12 to 20 from the guidelines and includes a mix of BASICS, LICENSING, and PLANT scenarios. This number of scenarios was arbitrarily chosen to be greater than 12 to 20 just to provide an example of the methodology for obtaining scenario data for three types of scenarios.

5.3 Summary

The scenario-generation method presented in Section 4.3 was successfully applied to both Systems A and B to determine the type and number per type of scenarios to be developed. However, the specification of the actual test scenarios did require detailed nuclear engineering knowledge and judgement. Such professional involvement will be required for the selection of specific validation scenarios for other nuclear AI systems or, indeed, for AI systems in other domains as well as conventional software. While some general guidelines were suggested for making these decisions, many issues will be domain-specific and can be evaluated only by an expert.

6 A NOVEL KNOWLEDGE-BASE SCENARIO-GENERATION METHOD

The new method described in this section should be extremely cost-effective in that it does not require the system to be actually operating but rather relies on an analysis of the knowledge-base to generate possible scenarios for review by a knowledgeable specialist. It should be noted that this method has been neither completely designed nor tested; its status is that of a potentially promising but untried technique.

The method is first contrasted with true validation scenario testing and described as it might appear in use (section 6.1). Then, in section 6.2, the principles underlying its performance are given. Section 6.3 concludes with a review of the merits of the KBSG approach.

6.1 General Description of Method

The validation scenarios discussed in the previous sections involve dynamic tests of a system -- setting up certain startup and data conditions, and then having the system under test perform in normal operation mode. In contrast, the method proposed here does not actually involve actual operation of the system. Rather, the method uses the knowledge-base component of the system under test (e.g., the rule-base or frames for expert systems, or the set of class and object-definitions for object-oriented systems). The knowledge-base is analyzed extensively by a separate program, and then scenario-like descriptions are generated for evaluation by someone familiar with the application domain.

The descriptions provided by the Knowledge-Base Scenario Generation (KBSG) method are, in their simplest form, a sequence of state-changes that the system could plausibly go through for some starting input-state and subsequent states of data and user input. For example consider a system which is supposed to monitor safety parameters and other data of a nuclear plant and, when there is an emergency, invoke the appropriate Emergency Operating Procedure (EOP) for the operator to follow, showing the state of the plant and the actions appropriate and suggested for that state. The EOPs involved in the system are taken directly from the hard-copy manual EOPs. Under a dynamic test of such a system, the system would be reading data directly from actual plant data-interfaces or from some kind of simulator, and the EOPs, when invoked, would be represented in a carefully-designed user-interface for operator processing. The operator would be given the plant status, the relevant EOP section and suggested action(s), and then asked for a decision. The operator would enter the decision, the system would implement it (or simulate its implementation), and then new data would eventually trigger another aspect of that EOP (or another EOP). This user-system interaction would continue until the plant was completely stable again.

If the KBSG method were used for the above situation, the operator (or some other user) would, generally, be looking at a display in which the initial status of the plant, the data-changes, the EOP, and the recommended actions were all described by a series of text statements rather than the nice user-interface display of the actual system. The system would not actually be running, and the KBSG session could actually be conducted almost on any mini-computer or personal computer located anywhere. The sequence of states presented to the user would not be determined by an actual plant or plant simulator. Rather, they would be possible states reachable from the prior conditions, and they would be selected according to some kind of simplified programmed heuristic. For example, the heuristic might arbitrarily decide to declare that the plant parameters are such that an EOP "RPV Control Procedure Entry" is entered; subsequently, it might select the plant conditions to invoke the "Primary Containment Control Procedure." Each decision taken by the KBSG would be accompanied by an explanation, stating in words the rules or other knowledge-base elements that justified such a decision. To continue with the example, at this point of +PC ENTRY

(Primary Containment section), the EOP embodied in the knowledge base calls for emergency depressurization actions (e.g., INITIATE HPCS SYSTEM, INITIATE LPCI-C, LOWER SUPPRESSION POOL TEMPERATURE BELOW 35 DEG C, etc.) which would be detailed for review by the user (and selected among if a choice is needed).

It is important to note that this method involves no plant simulation at all. Rather the program decided on one of the paths through the knowledge base; in a rule-based system, this will be a path from one rule to another until a final state is reached. Tho program will announce as the current state of the plant whatever are the IF conditions in the next rule on this path, and then it will announce the decision taken by that rule.

It is the user's task during the above KBSG scenario to review each sequential statement of plant-states, EOP invocations, offered explanations, and recommended operator actions and assess whether they are reasonable or not, whether they conform to the hardcopy EOPs, or whether the plant could ever be in such a state described. Anything suspect or deemed unreasonable should be flagged (perhaps by pressing a function key, entering a comment, etc., depending on how the KBSG was set up). This is very much the same task that is required when an actual validation scenario is dynamically executed on a real plant control environment, or on a simulator. The operator viewing the system performance is keeping a careful eye out for false, inconsistent, non-conforming, or otherwise puzzling outputs or actions of the system. The difference in the two cases is that, during actual execution of a validation scenario, the simulator or the actual plant will control the timing and value of the data parameters, while in the KBSG situation these will be generated or selected by a special program. In addition, of course, the KBSG user-interface will typically be impoverished relative to the actual system, since it is really not justified to dummy up a realistic user-interface for KBSG trials. However, with KBSG the operator will be given an explanation of each action of the system, according to the relevant rule, where this is typically only an option under normal operating conditions. Also, the KBSG user will explicitly be instructed to examine to see whether all of the IF conditions and THEN actions are necessary or whether any are missing.

6.2 Underlying Principles of Operation

The key to the operation of the KBSG method is analysis software which develops a representation of the system's knowledge for specific situations. Rule-based systems will be used to provide a detailed example, but the same principles will apply to frame-based, model-based, case-based, and object-oriented systems as well. It is important to note that the KBSG should be performed only on knowledge-bases that have been checked for obvious anomalies and errors, preferably by an automated checking program.

For a rule-based system the KBSG preliminary analysis software will develop a so-called rule-transition graph, showing what rules could be invoked by outcomes of what other rules, what data-states are essential for rules to be activated, and what the outcomes of each rule are. The graph is organized such that each node of the graph represents a unique rule (and has a unique identifier). For a particular node, the IF conditions of an IF-THEN rule are the inputs to the node, and the THEN actions (or assertions) are the outputs of the node. Most of the THEN outputs will be connected, as inputs, to subsequent rule nodes, but most nodes will also have data-inputs which do not come from previous rule firings but sample specific plant variables at that moment when the rule is being evaluated by the system to see if it can fire.

A scenario generated by KBSG is, very simply, the selection and description of one particular path through the rule-transition graph: the beginning input conditions are described; a choice of possible rule-nodes from that input set is made, and the IF conditions are output to the user as statements about the plant -- e.g., "RPV WATER_LEVEL CANNOT BE DETERMINED TO BE ABOVE LEVEL 1, LPCI-A IS NOT RUNNING, RHR-A IS NOT RUNNING IN CONTAINMENT SPRAY MODE ...". Then, the system would give the name of the rule-node and the actions called for under these conditions by the rule -- e.g., "Rule RC/L 1.8 fires under these conditions and calls for the following action: SET 3 III.B.1.b.3.a". The system would then describe a subsequent rule that is enabled by the preceding conditions, in the same fashion, continuing until a termination state is reached. Remember that this is very similar to what would be done under an actual scenario execution except that the user-interface would be fancier, and the explanation of the rule being invoked would not be present unless requested. However, the user would be given the recommendation of the THEN action (e.g., "SET 3 III.B.1.b.3.a") in both cases.

For small graphs, there are well-known algorithms that can generate a set of scenarios to "cover" the whole graph. The key issue for the KBSG method arises when the rule-graphs are quite large, with perhaps thousands of nodes; the question is how to select some reasonable number of inputs and paths through the graph that are likely to be of value in finding errors. One procedure is to develop metrics which attempt to assess the complexity of rules, and then pick the most complex rules to test. This can easily be automated, whatever the criteria for complexity are (e.g., number of IF conditions, number of negations, number of THEN actions, etc.; see Miller, 1990, for a discussion of such metrics.). Alternatively, those rule-nodes which are especially important, perhaps involving very critical safety situations, can be hand-selected by a human tester, and some number of paths leading from the primary input stage to that node can be automatically selected (such nodes can probably be located automatically if the selection criteria for "safety-criticality" or whatever else is of interest can be specified clearly). A particularly efficient scenario is one which follows a path through multiple "important" rule-nodes. Selection of scenarios by these procedures addresses the problem of there being actual explicit errors in either the IF or the THEN parts of the rule: the user, reading the descriptive text, can detect an incorrect comparison, an unknown variable reference, a nonsensical action, or any of a variety of other rule errors.

There are other kinds of possible errors in rules, particularly those of "over-generalization" and "over-specialization." The former refers to a set of IF conditions which are too broad and need one or more additional conditions to restrict the rule; the latter refers to a set of IF conditions which are too narrow, where one or more of the conditions need to be removed. The KBSG method can be adapted to explore these situations also in the following way. Assume that a particular path through the rule-graph has been chosen for other reasons. Over-generalization, for example, can be tested by adding to the IF conditions of a node one or more new plausible ones based on the following premise: if there are multiple rules which can be activated following a rule-node and several of these rules specify the same condition C, but C is not specified for the node on the chosen path, then <u>add the negation of this condition C</u> to the IF conditions. Users of the KBSG may be likely to detect that this negated condition is incorrect, but what is important is whether they say that the condition should be eliminated or changed to a positive state. If the latter, then it is possible that the rule was over-general and indeed required the same condition used by the other rules. Under-generalization can be tested in an analogous fashion. This procedure is very similar to that of "mutation testing" used for conventional software except that it is much more precise as to the specific defect being introduced (the "mutation") into the knowledge base.

Additional means of testing for rule errors can be achieved by similar rule augmentations, most of them quite automatable. An important remaining issue is how readable text is to be generated from the rule syntax, which may be quite unreadable in its correct form. The approach is to develop a special translator for each form of rule syntax which translates the rule parts into canonical forms. For example, an understandable canonical form for IF conditions is:

(VARIABLE COMPARISON-OPERATOR REFERENCE)

Given conditions such as "(LPCI-A IS NOT RUNNING)" and "(RHR-A PUMP Status EQUAL OFF)", then "LPCI-A" and "RHR-A PUMP Status" are seen to be the variables, "IS" and "EQUAL" are the comparison operators, and "NOT RUNNING" and "OFF" are the references. Another rule syntax might be: STATUS(RHR-A PUMP) = OFF. This would have to be translated into the same canonical form. Once the translator has been developed, then the IF conditions only need to be listed one after the other in this form, following such text as "The system is now found to be in the following state." Of course, it would also be quite reasonable to have the output given in audio form instead of visual if that were desirable (via a speech generator module).

The KBSG approach can also be developed for frame-based or object-oriented systems. For these systems, the algorithms which result in selection of frames for display or processing must be identified and made available for simulated operation by the KBSG module. Attributes and attribute-values will be displayed and changed in a manner similar to that described here for rule-based systems, and the user's task is again to identify any anomalies.

6.3 Merits of the KBSG Method

The reader should keep in mind that this method has actually neither been fully designed nor tested, and that the claims of automatability are therefore not shown. Nevertheless, were such a method as KBSG to be developed and automated, it would appear to have the following strong desirable features:

1) KSBG is concerned specifically with assessing the key component of expert systems, the knowledge-base, which is judged to be under-evaluated by conventional V&V methods.

2) It does not require a participant to directly examine the system's knowledge-base, which may be very difficult to comprehend for a non-expert. Rather, it requires only that the user follows and thinks about the description of a sequence of states and actions. Any application professional can therefore use this method to assess the knowledge-base.

3) It does not require dynamic execution of the whole system, usually in the context of an actual larger system or simulator. This avoids the cost and difficulties of scheduling a real validation scenario execution, assembling all the necessary persons to support the run, etc.

4) The information provided the participant of the KBSG method is typically richer, more focused and more informative than given a participant of an actual execution run. Therefore, the participant is more likely to detect problems.

5) The method can be run much more quickly, covering dozens or even hundreds more test conditions in the same time that it takes to run one condition in the standard dynamic-testing environment.

6) The easy possibility of adding or deleting conditions or actions from rules, or even of introducing changes within these, makes possible a variety of additional formats for assessing potentially problematic rules, such as presenting a multiple choice of rule alternatives and asking participants to select and justify their choice. Such a procedure could elicit much useful expert information.

For these reasons KBSG holds considerable promise as an additional technique for the testing of AI systems.

7 SUMMARY AND CONCLUSIONS

Activity 8 of this project entailed the development of a methodology for selecting AI systems validation scenarios. A comprehensive assessment of nuclear industry software users and managers by telephone and literature review resulted in an understanding of the current accepted state-of-the-art in conventional software validation scenarios. Validation scenarios were found to be an essential aspect to V&V in the nuclear industry, with a variety of different but consistent views on what the scenarios might look like. However, there are no standards, regulations, or guidelines concerning validation scenarios.

Six types of scenarios were identified and named: BASICS, PLANT, TEST, REGRESSION, CODE, and LICENSING. They were each analyzed for their relative benefits and limitations. Thirty-nine actual cases of nuclear industry software with their associated number and type of validation scenarios were evaluated. Discussions were held with software quality assurance engineers and managers at a variety of nuclear industry organizations including utilities, consultants, architect-engineers, and nuclear steam supply system vendors. Choice of type of scenario was found to depend upon their availability, applicability, fidelity, and importance to the specific system.

Validation scenario selection guidelines were developed which are based on the V&V Class and nature of the software. These guidelines set a minimum number of scenarios and type of scenario along with a recommended order of importance of the scenario types. The guidelines were applied to System A and B resulting in a recommended set of scenarios which should be run on each system. A concept for a Knowledge-Based Scenario Generator was presented which could greatly facilitate the testing of such types of AI systems with validation scenarios.

8 REFERENCES

ANSI/ANS-3.5-1985, *American National Standard Nuclear Power Plant Simulators for Use in Operator Training*, American Nuclear Society, 555 North Kensington Ave., La Grange Park, Illinois, 60525, October 25, 1985.

ASME NQA-2a-1990 Part 2.7, *Quality Assurance Requirements of Computer Software for Nuclear Facility Application*, The National Institute of Standards and Technology Computer Systems Laboratory, Gaithersburg, Maryland 20899, 1990.

Baltimore Gas and Electric Company, *A-85-11A, Toopical Report - RETRAN Computer Code Reactor System Transit Analysis Model Qualification*, Baltimore Gas and Electric Company, January 31, 1986.

Chaffee, A.E., Chief of Events Assessment Branch, Division of Operating Reactor Support, USNRC, *Listing of Augmented Inspection Teams Dispatched Since 1985*, June 8, 1993.

Chang, W.C., and J.F. Cheng, *The Utility Experience of Implementing the Emergency Operating Procedure Tracking System*, Proceedings of Applications for the Electric Power Industry Conference, Electric Power Research Institute, Palo Alto, California 94303, June 1990.

Code of Federal Regulations, Title 10, Part 50.73, *Licensee Event Report System*, January 1, 1992.

EPRI NP-1850-CCM VOLUME 4, *Quality Assurance Requirements of Computer Software for Nuclear Facility Application*, Electric Power Research Institute, Palo Alto, California, 1990.

EPRI NP-2511-CCM VOLUME 4, *RETRAN-02 - A Program for Transient Thermal-Hydraulic Analysis of Complex Fluid Flow Systems - Volume 4 Applications*, Electric Power Research Institute, Palo Alto, California, January 1983.

EPRI NP-5524, *Testing and Installation of a BWR Digital Feedwater Control System, Atomic Energy of Canada, Limited*, Electric Power Research Institute, Palo Alto, California, December 1987.

Laue, S.A., *Technical Specifications Advisor Pilot Project for Brunswick Steam Electric Plant - Unit 1 Carolina Power and Light Company*, Conference on Expert Systems Applications for the Electric Power Industry, Orlando, Florida, June 5-8, 1989.

Miller, L.A., *Dynamic Testing of Knowledge Bases Using the Heuristic Testing Approach. Expert Systems with Applications: An International Journal*, Special Issue: Verification and Validation of Knowledge-Based Systems, Vol. 1, No. 3, pp. 249-269, 1990.

Miller, L.A., J. Hayes, and S. Mirsky, *Task 7: Guidelines for the Verification and Validation of Artificial Intelligence Software Systems*, Report prepared for United States Nuclear Regulatory Commission and the Electric Power Research Institute, May 1993.

NSAC-40, *Accident Sequences for Design, Validation, and Training-Safety Parameter Display System*, Science Applications International Corporation for Nuclear Safety Analysis Center, Nuclear Safety Analysis Center, Atlanta, Georgia, April 1982.

NSPSAD-8102NP, *Reload Safety Evaluation Methods for Application to PI Unites*, Northern States Power Company, December 1981.

NTH-G-6, *Topical Report - RETRAN Code Transient Analysis Model Qualification*, Florida Power & Light Company, July 1985.

NUREG-0800, *USNRC Standard Review Plan, Chapter 15*, revised July, 1981.

NUREG-1272, *Annual Report*, USNRC Office for Analysis and Evaluation of Operational Data 1991, Vol. 6, No. 1, July 1992.

NUREG/CR-4674, *Vol. 15 and 16, Precursors to Potential Severe Core damage Accidents: 1991 A Status Report*, September 1992.

NUREG/CR-4195, *Overview of TRAC-PD2 Assessment Calculations*, EG&G Idaho, November 1985.

NUREG/CR-4196, *Overview of TRAC-BD1(Version 12) Assessment Studies*, EG&G Idaho, April 1985.

NUREG/CR-4454, *RELAP5/MOD2 Code Assessment at the Idaho National Engineering Laboratory*, March 1986.

NUREG/CR-4428, *Overview of TRAC-BD1/MOD1 Assessment Studies*, EG&G Idaho, November 1985.

NUREG/CR-4674, *Vol. 15 and 16, Precursors to Potential Severe Core damage Accidents: 1991 A Status Report*, September 1992.

NUSCO 140-1, *NUSCo Thermal Hydraulic Model Qualification Volume I (RETRAN)*, Northeast Utilities Service Company, August 1, 1984.

SNUPPS (Standardized Nuclear Power Plant System), *Steam Generator Single-Tube Rupture Analysis for SNUPPS Plants Callaway and Wolf Creek*, December, 1985.

TVA-TR81-01, *BWR Transient Analysis Model Utilizing the RETRAN Program*, Tennessee Valley Authority, December 31, 1981.

U.S. Nuclear Regulatory Commission Regulatory Guide 1.70, *Format and Content of Safety Analysis Reports for Nuclear Power Plants*, 1982.

U.S. Nuclear Regulatory Commission Safety Evaluation Report, *Reactor Physics and Reload Safety Evaluation Methods Technical Reports NSPNAD-8101P and -8102P for the Northern States Power Company Prairie Island Nuclear Generating Plant, Unit Nos. 1 and 2*, February 22, 1983.

U.S. Nuclear Regulatory Commission Safety Evaluation Report, *Main Steam Line Break Analysis - Yankee Nuclear Power Station*, November 30, 1983.

UAI 83-15, *Qualification of DYNODE-P for Anticipated Transients With and Without SCRAM Using LOFT Tests*, Utility Associates International (service of Control Data Corporation), April 1, 1983.

VEP-FRD-19, *The PDQ07 Discrete Model*, Virginia Electric and Power Company Richmond, Virginia, July 1976.

VEP-FRD-20, *The PDQ07 One Zone Model*, Virginia Electric and Power Company, Richmond, Virginia, January 1977.

VEP-FRD-23, *Reactor Core Thermal Hydraulic Analysis Model Using LYNX1 and LYNX2 Computer Codes*, Virginia Electric and Power Company, Richmond, Virginia, August 1978.

VEP-FRD-24, *The VEPCO FLAME Model*, Virginia Electric and Power Company, Richmond, Virginia, October 1978.

VEP-FRD-33, *Reactor Core Thermal-Hydraulic Analysis using the COBRAIIIC/MIT Computer Code*, Virginia Electric and Power Company, Richmond, Virginia, August 1979.

VEP-FRD-41, *Reactor System Transient Analyses using the RETRAN Computer Code, Virginia Electric and Power Company*, Richmond, Virginia, March 1981.

VEP-NFE-2, *VEPCO Evaluation of the Control Rod Ejection Transient*, Virginia Electric and Power Company, Richmond, Virginia, October 1983.

Wisconsin Public Service Corporation, *Reload Safety Evaluation Methods for Application to Kewaunee.*

Wisconsin Public Service Corporation, *Qualification of Reactor Physics Methods for Application to Kewaunee*, September 29, 1978.

NRC FORM 335
(2-89)
NRCM 1102,
3201, 3202

U.S. NUCLEAR REGULATORY COMMISSION

BIBLIOGRAPHIC DATA SHEET

(See instructions on the reverse)

1. REPORT NUMBER
(Assigned by NRC. Add Vol., Supp., Rev., and Addendum Numbers, if any.)

NUREG/CR-6316
SAIC-95/1028
Vol. 6

2. TITLE AND SUBTITLE

Guidelines for the Verification and Validation of Expert Software and Conventional Software

Validation Scenarios

3. DATE REPORT PUBLISHED

MONTH	YEAR
March	1995

4. FIN OR GRANT NUMBER

L1530

5. AUTHOR(S)

S.M. Mirsky, J.E. Hayes, L.A. Miller

6. TYPE OF REPORT

7. PERIOD COVERED *(Inclusive Dates)*

8. PERFORMING ORGANIZATION — NAME AND ADDRESS *(If NRC, provide Division, Office or Region, U.S. Nuclear Regulatory Commission, and mailing address; if contractor, provide name and mailing address.)*

Science Applications International Corporation
1710 Goodridge Drive
McLean, VA 22102

9. SPONSORING ORGANIZATION — NAME AND ADDRESS *(If NRC, type "Same as above"; if contractor, provide NRC Division, Office or Region, U.S. Nuclear Regulatory Commission, and mailing address.)*

Division of Systems Technology
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, DC 20555-0001

Nuclear Power Division
Electric Power Research Institute
3412 Hillview Avenue
Palo Alto, CA 94303

10. SUPPLEMENTARY NOTES

11. ABSTRACT *(200 words or less)*

This report is the sixth volume in a series of reports describing the results of the Expert System Verification and Validation (V&V) project that is jointly funded by the U.S. Nuclear Regulatory Commission and the Electric Power Research Institute to develop guidelines for the V&V of expert and other systems. This activity was concerned with the development of a methodology for selecting "validation scenarios." These are defined as "realistic dynamic tests of software which covers only the intended range of applications of the software and are designed to sample important subsets of functions, usually for selected situations known to be challenging or problematic, to provide assurance that the system achieves the tested functions with the required accuracy and performance." Such scenarios are used after all the V&V testing of the system is completed. Five categories of validation scenarios were defined: PLANT, TEST, BASICS, CODE, and LICENSING. A sixth type, REGRESSION, is a composite of the others and refers to the practice of using trusted scenarios to ensure that software modifications did not unintentionally change non-modified functions. A generalized procedure was developed for generating apprporiate sets of validation scenarios from these basic categories.

12. KEY WORDS/DESCRIPTORS *(List words or phrases that will assist researchers in locating the report.)*

validation, verification, V&V expert systems, knowledge base, guidelines, scenarios, software quality assurance

13. AVAILABILITY STATEMENT

Unlimited

14. SECURITY CLASSIFICATION

(This Page)

Unclassified

(This Report)

Unclassified

15. NUMBER OF PAGES

16. PRICE