SANDIA REPORT
SAND2000-3101
Unlimited Distribution
Printed January 2001

Guidelines for Sandia ASCI Verification and Validation Plans – Content and Format: Version 2.0

Martin Pilch, Timothy Trucano, Jaime Moya, Gary Froehlich,
Ann Hodges, David Peercy

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,
a Lockheed Martin Company, for the United States Department of
Energy under Contract DE-AC04-94AL85000

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

SAND2000-3101
Unlimited Release
Printed January 2001

Guidelines for Sandia ASCI Verification and Validation Plans — Content and Format:

Version 2.0

Martin Pilch
Validation and Uncertainty Methodologies

Timothy Trucano
Optimization and Uncertainty Estimation

Jaime L. Moya
Nuclear Weapons Program Support

Gary Froehlich
Distributed Information Systems

Ann Hodges
Advance Decision Support Applications

David Peercy
Quality Engineering

Sandia National Laboratories
P. O. Box 5800
Albuquerque, New Mexico 87185-0819

Abstract

This report updates general guidelines for the development of verification and validation (V&V) plans for ASCI code projects at Sandia National Laboratories. The main content areas recommended by these guidelines for explicit treatment in Sandia V&V plans are (1) stockpile requirements; (2) key phenomena to be modeled by the individual code and PIRT; (3) software quality engineering; (4) physics and engineering verification test plan; and (5) code application specific validation test plan. The authors of this document anticipate that the needed content of the V&V plans for the Sandia ASCI codes will evolve as time passes. These needs will be reflected by future versions of this document.

(Page Left Blank)

Table of Contents

List of Figures

(Page Left Blank)

List of Acronyms

ASCI	Accelerated Strategic Computing Initiative
AWLPG	Albuquerque Work Load Planning Guidance
BEU	Best Estimate plus Uncertainty
CFD	Computational Fluid Dynamics
DOE	Department of Energy
DP	Defense Programs
M&S	Modeling and Simulation
NS	National Security
PIRT	Phenomena Identification and Ranking Table
SBSS	Science Based Stockpile Stewardship
SLEP	Stockpile Life Extension Program
SQE	Software Quality Engineering
SSP	Stockpile Stewardship Program
V&V	Verification & Validation
VERTS	Verification Test Suite
VALTS	Validation Test Suite

(Page Left Blank)

1. Introduction

The Department Of Energy's (DOE's) Accelerated Strategic Computing Initiative (ASCI) is designed to develop high performance computational tools and models to help manage the safety and reliability of the enduring nuclear stockpile. It is intended that the resulting computer codes be applied to high consequence problems. An important element of high consequence modeling and simulation (M&S) is a sound and viable verification and validation (V&V) program – one that will substantively increase the credible predictive content of M&S for Science Based Stockpile Stewardship (SBSS) while remaining within the constraints of available funding resources.

Verification and validation are defined in the DOE Defense Programs (DOE/DP) ASCI Program Plan [DOE-2000a] as:

> **Verification** – The process of determining that a computational software implementation correctly represents a model of a physical process.
>
> **Validation** – The process of determining the degree to which a computer model is an accurate representation of the real world from the perspective of the intended model applications.

These definitions are equivalent to definitions promulgated by AIAA [AIAA] and the Defense Modeling and Simulation Office [DMSO]. More informal definitions are also applicable in physical science based M&S and intuitively understood by workers performing ASCI code development and code application. They are:

> **Verification** – The process of determining that the equations are solved correctly.
>
> **Validation** – The process of determining that the equations are correct.

These latter definitions recognize the basic focus of ASCI M&S, which essentially deals with the numerical solution of equations. At the same time, these definitions emphasize that V&V is an open-ended process. It is particularly desirable to view V&V as a quality improvement process. This view underlies our approach in this document.

The purpose of the present document is to establish guidance for developing V&V plans for ASCI code projects at Sandia National Laboratories. It is an update of the first version published in 1999 [Trucano]. V&V activities are a critical component of the process for establishing confidence in applying a code to a specific application. Developing detailed V&V plans contributes to the achievement of both the code M&S requirements and the programmatic needs of Sandia's National Security (NS) sector, as defined, for example, in the Stockpile Life Extension Plan (SLEP). V&V plans are expected to demonstrate and document how the quality and fidelity of the code M&S capability will satisfy specific stockpile programmatic requirements. Each code team, in partnership with their stockpile customers and designers, is expected to create, maintain, and execute these plans throughout the life cycle of the code.

The V&V plans developed and maintained for Sandia ASCI codes will also be reviewed to ensure that these plans have sufficient breadth and depth to establish needed confidence in the intended application of the codes. This review process, called ***peer review***, has been defined in [Pilch]. The nuclear weapons design community at Sandia will be included in review activities for code V&V plans. As described in [Pilch] DP at Sandia is a partner in the V&V peer review process.

The remainder of this document is organized as follows. In Section 2 we discuss overall constraints placed on code V&V plans by the Sandia ASCI V&V program. The most basic paradigm for V&V at Sandia is requirements-based V&V. The need for linking DP customer requirements to V&V activities is discussed in Section 3. We have selected the Phenomena Identification and Ranking Table (PIRT) as the device for translating Sandia DP requirements into M&S tasks that drive V&V activities for specific codes. We describe the PIRT process and results in Section 4.

Software Quality Engineering (SQE) addresses code verification from a more comprehensive viewpoint than simply software testing. Guidance on the proper identification of SQE activities as part of V&V activities is now derived from DOE HQ guidance [DOE-2000b]. In Section 5 we summarize this guidance and the impact it makes on the content of Sandia V&V plans. Because of the unique importance of verification testing of the physics and engineering mathematical models in scientific computer codes, we have also defined content for separate discussion of verification testing in Section 6. Required content for planning code validation activities is given in Section 7. The need for V&V plan content that develops guidance for appropriate use of the code for the targeted stockpile applications is discussed in Section 8.

Throughout this document, the major required V&V plan contents are summarized in the form of ***content criteria***. These criteria were not specifically defined in Version 1 of these guidelines. Such criteria are provided in this version to increase specific guidance, as well as to provide traceable content metrics for the V&V peer review process. Appendix A provides an outline of the required Table of Contents for Sandia V&V plans, with the content criteria properly located.

We recommend that the reader of these guidelines be familiar with the first version published in 1999 [Trucano]. We also recommend to the reader J. Lee's Sandia report [Lee].

2. General Guidance for V&V Plans

2.1 Approach

The main V&V guidelines presented in this document are summarized as follows. The code V&V plan will be written to:

- Identify a specific stockpile driver that focuses V&V activities.

- Demonstrate understanding of the customers associated with the chosen stockpile driver and their M&S needs. Customer requirements associated with the selected driver are typically defined in such plans as the Stockpile Life Extension Program (SLEP), the Albuquerque Work Load Planning Guidance for various years (AWLPG 99-0 for 1999, for example), the weapons system groups' work definitions, and other DP weapons program guidance available at Sandia.

- Translate the critical DP requirements associated with the stockpile driver into prioritized M&S requirements. The M&S requirements thus determined must be sufficient for defining V&V activities that are necessary to achieve confidence in the targeted application of the M&S. The PIRT methodology is the chosen means of performing this task and must be developed in the V&V plan.

- Demonstrate understanding of the role of SQE in increasing confidence in the code M&S.

- Describe the focused physics/engineering verification testing to be performed during execution of the V&V plan.

- Describe the validation testing to be performed, including experimental requirements and priorities, during execution of the plan.

- Provide guidance for the proper use of the code M&S for critical stockpile applications. Such guidance must evolve through collaboration with the identified DP customers for the code M&S capability.

Each of these content areas will be discussed in greater detail in Sections 3 through 8 of this report.

Each plan will be written for a specific major stockpile driver. If a given code has several stockpile drivers, then individual plans for each driver should be developed in the course of time. However, planning activities are obviously prioritized by the needs of DP at Sandia. For codes with DP mileposts on their schedules, this is clearly an opportunity for coordination as well as a planning constraint. For example, the specific stockpile driver chosen for focusing the first V&V plan should logically align with schedule and priority-weighted milepost requirements arising from DP programs at Sandia.

There are many questions that influence the development of a requirements-based V&V plan having content like that described above. A few examples of relevant questions are:

- What is the current development status of the code M&S capability?
- What are the most important stockpile drivers underlying the code project?
- What are the gaps between the present status of the code M&S capability and what is required to perform appropriate V&V activities?
- What are V&V planning assumptions and how do they correlate with the M&S requirements originating in the stockpile driver?
- What technical barriers to success of the V&V activities have been identified?
- What will be key performance indicators that measure success of V&V activities?

These and other questions must be implicitly and explicitly answered if a code V&V plan has been successfully designed. It is impossible for a single document, such as the current guidelines, to provide complete guidance for the development of effective V&V plans for all ASCI codes under development at Sandia. Our goal, instead, is to emphasize content areas that are important, yet general enough to cut across all Sandia codes. The requirements-based V&V paradigm is the most appropriate context in which to achieve this goal.

2.2 Sandia ASCI V&V Programmatic Requirements

Each code V&V plan will conform to the following general requirements of the Sandia V&V program.

2.2.1 Ownership and Authors

The V&V plan will have a clearly defined owner, presumably the lead author of the plan. It is understood, unless otherwise stated, that the plan owner is also the individual with primary responsibility for overseeing the implementation and execution of the code V&V plan. In addition, the list of authors for the code V&V plan will include (1) an experimenter familiar with key experimental validation issues; (2) a representative of the DP customer that owns the stockpile driver the V&V plan is being written for.

2.2.2 Uniformity

All Sandia code V&V plans will be structured similarly, as well as have the same common elements. The proper definition for this plan structure is provided in this report. Appendix A summarizes the desired content and its order of presentation in the plan. The order suggested in the appendix is precisely the order of presentation in this report. Uniform organization of Sandia V&V plans promotes ready access to information across the Sandia ASCI program. This eases the sharing of information, as well as the tracking of progress. In addition, uniformity facilitates peer review, a key element in the Sandia V&V program, as well as other independent assessments that the ASCI program may be subject to in the future.

2.2.3 Focus on stockpile drivers

Stockpile drivers (discussed in greater detail in Section 3), defined in the AWLPG and determined through iteration with the Sandia weapon design community, will govern the V&V program at Sandia and the associated processes. Each code's V&V plan will be aimed at a specific stockpile driver. This substantially limits the scope of the requirements that are expected in the stockpile requirements and PIRT (see below) sections of this document. Additional stockpile drivers are to be included in subsequent plans, if necessary. The importance of connection to and communication with the Sandia weapons program is clearly emphasized by this approach.

2.2.4 Quality improvement

Each code V&V plan should be viewed as a rational quality improvement process for the code. The ultimate goal of the Sandia V&V program, and individual code V&V plans, is to improve the ability of the Sandia ASCI codes to achieve the M&S objectives of their stockpile drivers. This result should be echoed in increased confidence in applying the code. This view leads to two important elements for a Sandia V&V plan.

- First, it is important to describe the process, or processes, that govern the ***maintenance*** and possible future ***modification*** of the code V&V plan. In particular, how given planning content elements are to be maintained and modified should be described. It is also important to describe how V&V information artifacts will be generated and maintained. Finally, the roles and responsibilities of the members of the V&V team should also be clearly defined.

- Second, success of a code's V&V plan must be determined by developing and applying explicit measures of success, failure, and improvement. These measures, called ***metrics*** for short, must ultimately be connected to the stockpile driver requirements and their priorities.

2.3 Documentation Tree

The V&V content discussed in this report is so great that a single document that expresses it would likely be too large for effective use by the code team or assessment by peer review panels. We encourage the code teams to view the "V&V Plan" in many of the requested content areas as a summary and guidance document. In this role, the V&V plan should provide an overview of a larger system of documentation, while communicating the principles and logic by which the plan has been formulated and pointing to additional evidence. The plan should also stress the content areas of the stockpile driver and its requirements, as well as the resultant PIRT. These are the key elements from which everything else follows. It is unlikely that these content components will be documented in detail in any other form than the V&V plan. Figure 1 provides one view of the way a documentation tree could be arranged to detail the content elements expected in Sandia V&V plans.

2.4 Content Criteria Summary

The following summarizes the key content criteria reflecting the V&V plan elements discussed in this section.

PR1: The V&V plan authorship includes the V&V process owner, and experimenter, and a DP customer representative.

PR2: The plan is compatible with the format specified in the Version 2.0 guidelines.

PR3: A single stockpile driver for the V&V plan is identified. (This is reinforced in Section 3.)

PR4: The V&V planning process is described.

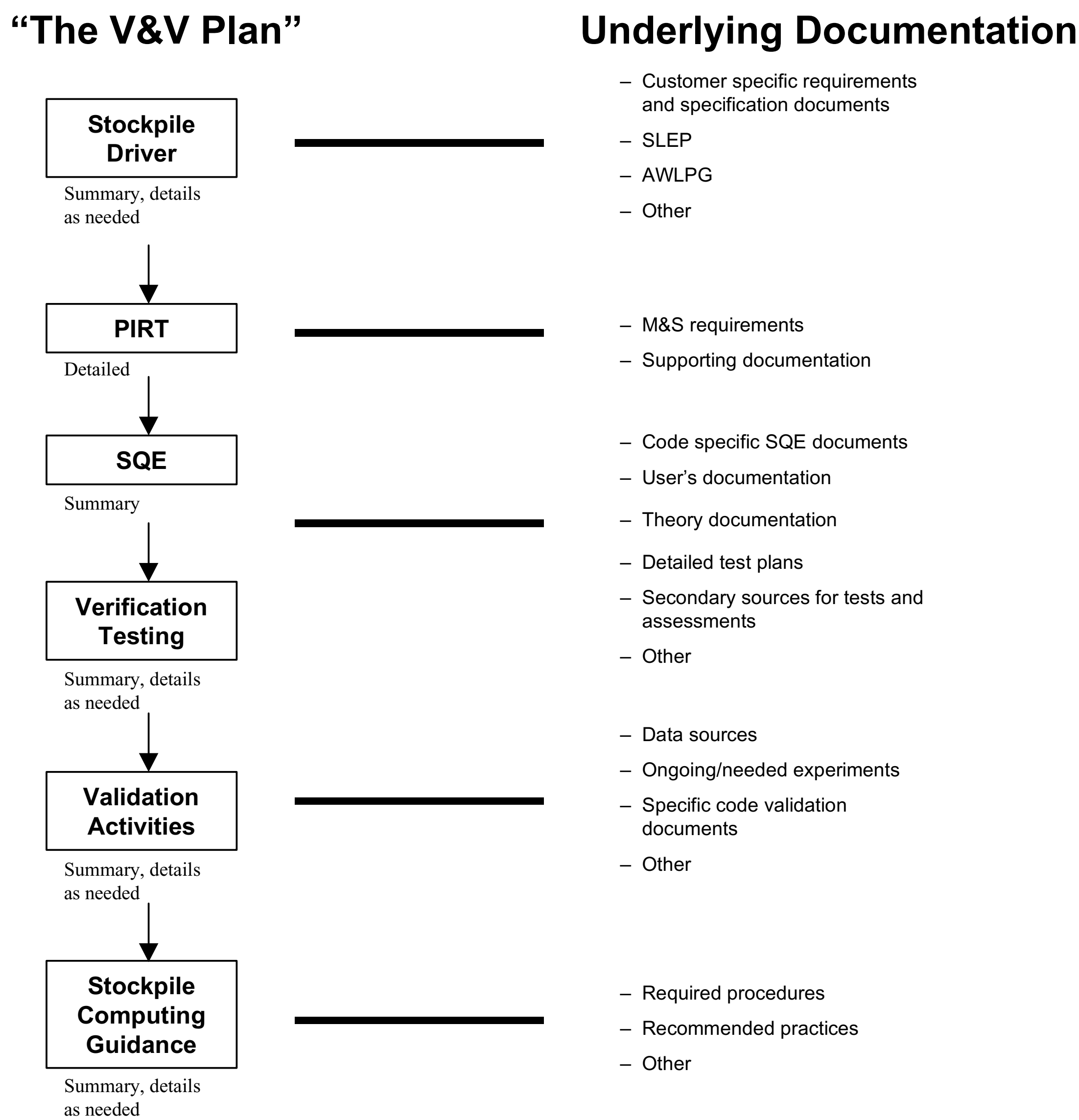

Figure 1. Possible documentation tree for Sandia V&V plans.

(Page Left Blank)

3. Stockpile Drivers, Needs, and Requirements

3.1 The Need for Requirements

The DOE/DP's ASCI Program Plan [DOE-2000a] describes the vision for verification and validation as establishing "confidence in the simulations supporting the Stockpile Stewardship Program through systematic demonstration and documentation of the predictive capability of the codes and their underlying models." To achieve this vision, each Sandia ASCI code's V&V activities must be fundamentally and explicitly driven by stockpile requirements. Therefore, precisely defining the appropriate stockpile requirements for each code project is paramount.

The stockpile requirements that govern the Sandia ASCI applications codes are found in plans such as the Stockpile Life Extension Plan (SLEP), Albuquerque Work Load Planning Guidance (AWLPG 99-0), and the detailed needs of weapon system design groups. These requirements determine the objectives that influence the subsequent development of code M&S requirements. These requirements, in turn, drive the particular V&V activities called for by a V&V plan. In the following, we refer to the core stockpile requirements that directly influence the code M&S requirements and associated V&V activities of a Sandia ASCI code project as ***stockpile drivers***.

The flow of information that begins with the specification of stockpile drivers and ends with V&V activities for a code is illustrated in Figure 2. The important thing to emphasize about this figure is that in requirements-based V&V all of the critical V&V process activities originate in the stockpile requirements that define the M&S requirements. An accurate understanding of stockpile drivers and the translation of this understanding into code M&S requirements are of critical importance for the definition and success of the V&V plan for a code.

The tool used for creating V&V activities from M&S requirements shown in Figure 2 is the ***Phenomena Identification and Ranking Table*** (***PIRT***), which will be discussed in detail in the next section.

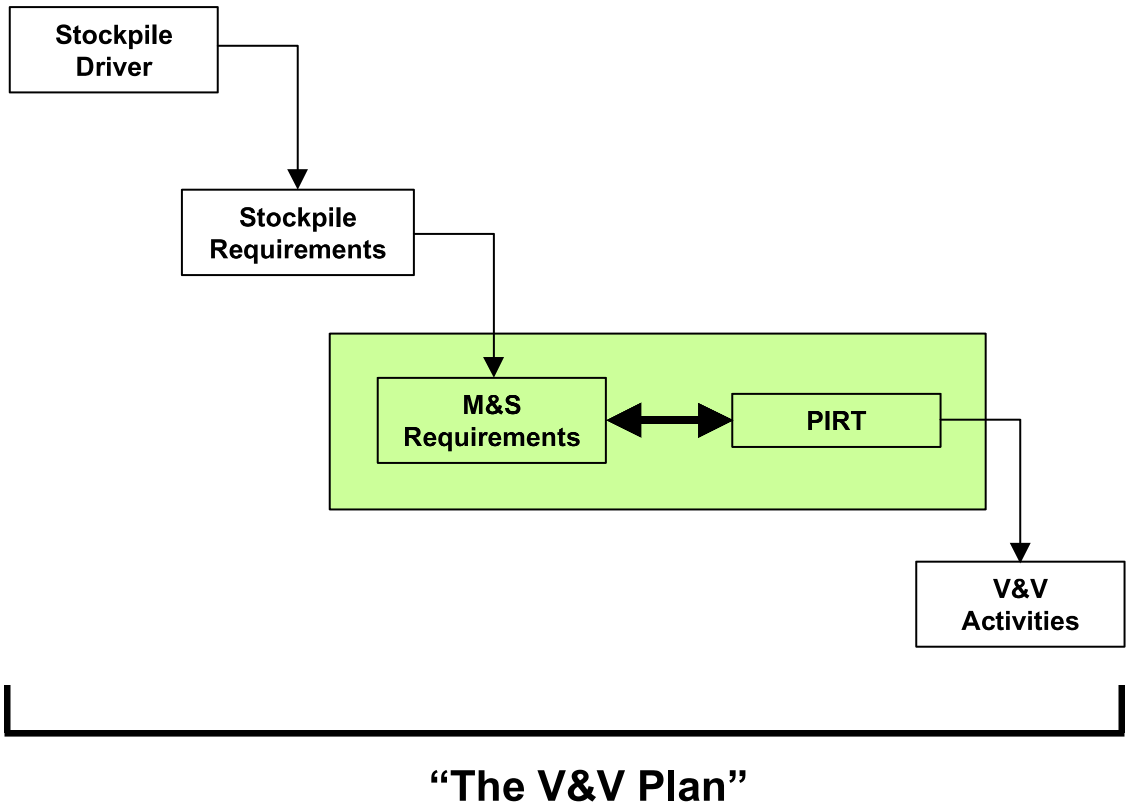

Figure 2. Flow of requirements to produce V&V activities.

3.2 Single Stockpile Driver

The development of stockpile and M&S requirements, hence development of the PIRT and planning of V&V activities, are completely dependent upon the selection of the stockpile driver. Most codes under development at Sandia have multiple stockpile drivers. This is fundamental to the nature of general-purpose computational science codes. However, validation is dependent on specific applications. The Sandia V&V program emphasizes that the proper target of validation activities is to specific applications of codes, not to codes. As discussed in Section 2, to provide this emphasis the V&V plan will therefore focus on one stockpile driver.

The choice of a single stockpile driver is a balance between generality and specificity. If the chosen driver is too general the resulting V&V plan will likely fail to develop a PIRT in sufficient detail. This will result in V&V activities that will be hard to coordinate and harder to prioritize. If the chosen driver is too specific the PIRT and resulting V&V activities will also be too specific. This will prevent reasonable leveraging of V&V activities to yield the largest sensible scope for the V&V plan. Either extreme will rapidly increase the costs, especially time, associated with V&V planning and make it more difficult to accomplish more than incremental impact under the stated plan.

In addition, very specific and narrowly focused validation activities are more akin to code qualification activities. (We will comment on this further in Section 8.) While we believe that code qualification for very specific code applications is a necessary consequence of rigorous V&V, qualification should follow from V&V efforts, not define them.

An example of a stockpile driver which is too broad is: "Model the transient dynamics of gravity bombs." An example of a stockpile driver that is probably too narrow is: "Apply PRESTO to model the transient dynamics of the XYZ bomb upon impacting ABC material at LMN velocity." A stockpile driver which has about the right scope for generating reasonable detailed requirements is: "Apply PRESTO to model the transient dynamics of the XYZ bomb in lay-down scenarios for a specified set of target materials." While nothing in the last statement immediately translates into M&S requirements for PRESTO, it is clear that detailed stockpile requirements can be uncovered, translated into M&S needs, and then used to create a PIRT and drive V&V activities. When completed, it is likely that the resulting V&V plan will also be applicable in large part to similar scenarios for the X1Y1Z1 bomb as well. This, in a nutshell, is the intended art of proper choice of a stockpile driver.

Individual code teams will vary considerably in terms of the effort expended in developing the stockpile and M&S requirements, the amount of information available at the start of the planning process, and how sensitive the V&V plan will be to variability in the DP programs at Sandia. Each team's written plan will mirror the level of effort expended to understand and document the customers' requirements at the time of writing.

3.3 DP Collaboration

Understanding the stockpile drivers and devising detailed stockpile requirements must be done in coordination with Sandia DP personnel (e.g. nuclear weapons design community, weapon systems groups, and NWSBU personnel) who are stakeholders in the specific code development project. A suitable strategy must identify these stakeholders, although the correct approach for doing this will vary with the code project. Detailed requirements for both code models and code software should follow from good understanding of the stockpile requirements.

When developing the needed requirements from a choice of stockpile driver, one must answer questions of the following kind.

- <u>What</u> stockpile driver, associated DP scenarios, and critical DP milestones does the code V&V plan intend to support?

- <u>Who</u> are the DP customers for the stated stockpile driver?

- When must the code be capable of supporting the important milestones in the weapons program directed at the stated stockpile driver?

- What is the intended impact of the code on DP for the stated stockpile driver?

- What previous DP development or testing activities will be supplemented or replaced by the code if the code is qualified for application to the important scenarios?

It is hard to imagine these questions being answered without close involvement of knowledgeable DP personnel in the planning process that results in V&V activities. It is for this reason that the V&V program has emphasized that a DP customer participate on the author team for the plan (see content criterion PR1).

3.4 Content Criteria Summary

The following summarizes the key content criteria reflecting the V&V plan elements discussed in this section.

PR3:	A single stockpile driver for the V&V plan is identified. (See Section 2)
DP1:	The appropriate customers and constraints associated with the stockpile driver have been identified.
DP2:	The detailed stockpile requirements have been extracted from the stockpile driver.
DP3:	The stockpile requirements have been mapped to M&S needs and requirements.
DP4:	There requirements are sufficient to allow the development of a useful PIRT.

4. Phenomena Identification and Ranking Table

4.1 Introduction

Stockpile requirements ultimately govern the fidelity requirements of the code. The fidelity requirements must be defined and assessed in terms of the identification, mathematical formulation, and software implementation of the key physical phenomena modeled by the code. While stockpile drivers and the related requirements are *necessary* for this task they are not *sufficient*. A process is needed to refine stockpile drivers into a more sufficient set of requirements that can serve as the basis for V&V activities.

The ***Phenomena Identification and Ranking Table (PIRT)*** is the methodology by which the key physical phenomena are defined. The PIRT ranks the importance of each code activity associated with implementing the phenomena and provides the basis for gauging associated fidelity requirements. The PIRT is an intermediate and necessary step for developing code V&V plans. The PIRT is a logical mapping between stockpile and M&S requirements and prioritized V&V activities.

The PIRT represents in three respects the refined model requirements that result from stockpile requirements. First, the PIRT identifies a set of needed physical phenomena to which code V&V requirements directly map. Second, the PIRT prioritizes the relative importance of the needed physical phenomena to the DP modeling and simulation objectives of the code. Third, the PIRT measures the current and future ability of the code to accurately represent and implement the needed physical phenomena.

The PIRT systematically identifies physical phenomena required for the modeling and simulation needed by the chosen stockpile driver. These are the phenomena the code must ultimately address, formulate, and implement to succeed in its stockpile mission. The phenomena must be prioritized and the criteria applied to accomplish this must be described. Finally, the adequacy or inadequacy of current models of the phenomena and their implementation must be discussed.

4.2 What is the PIRT?

The PIRT process originated in the needs of the U. S. nuclear power industry for high-consequence M&S. As discussed in [Wilson-1998] the PIRT process was devised to directly support the "***Best Estimate plus Uncertainty***" (BEU) approach for using computational modeling in nuclear reactor safety analyses (and subsequently waste repository performance assessment). The ASCI program has also applied a concept with some similarity to the PIRT. The ***Simulation Development Roadmap*** [Larzelere] is also intended to prioritize needed simulation capability development required to support specific application objectives for ASCI codes.

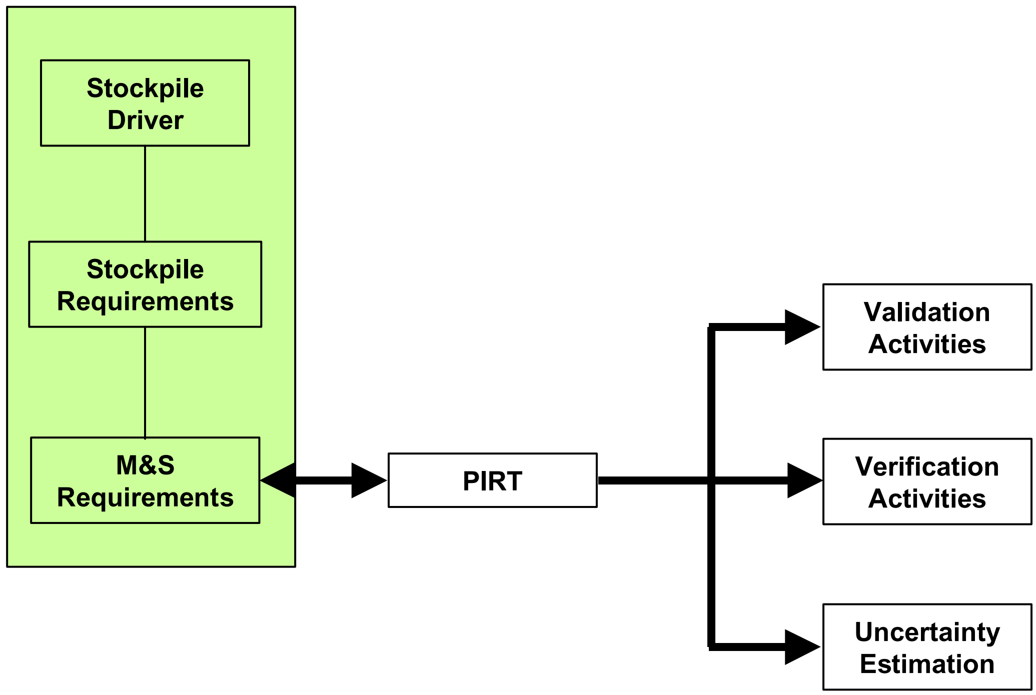

Figure 3. Conceptual role of the PIRT.

[Wilson-1998] argues that the PIRT has direct impact on three critical issues associated with the role of M&S in reactor safety assessment. First, it defines experimental priorities that provide confidence in the M&S. Second, by prioritizing phenomena, the PIRT serves to influence the progress of code development. Third, the phenomena priorities also directly influence the assessment of M&S uncertainty.

We present in Figure 3 a modification of a diagram in [Wilson-1998] that expresses these outcomes of the PIRT in a context that is more specific to V&V. As shown in that figure the PIRT has impact on three V&V-related content areas – validation (including the validation experiments that were the focus of Wilson); verification; and uncertainty estimation.

The impact on validation activities is natural, as validation is implicitly dependent on phenomena prioritization and fidelity requirements. Impact on uncertainty estimation is also an outcome of the PIRT process. This outcome is most strongly associated with stockpile computing (for example, the need to predict design margins). But, uncertainty estimation is also directly important in validation activities. That the PIRT also influences verification activities, in particular physics/engineering verification testing is clear. We will briefly illustrate the potential use of the PIRT in guiding verification testing in Section 4.3 below, as well as discuss it further in Section 6.

The PIRT process is part of a larger methodology used in the nuclear power industry to support the M&S paradigm of BEU. This methodology is called the ***Code Scaling, Applicability, and Uncertainty*** (CSAU) methodology. This larger methodology is discussed at length in [TPG]. A set of journal articles were developed from this report and published in 1990, including [Boyack], [Wilson-1990], and [Wulff]. More recent papers that discuss CSAU are [Wilson-1998] and [Zuber]. It is likely that the BEU M&S paradigm will be relevant to future stockpile computing at Sandia using ASCI codes, so the CSAU methodology is of interest beyond its utilization of PIRTs.

We will provide some general guidance on developing PIRTs to support V&V activities in Section 4.3 below. This discussion by no means replaces examination of the existing literature on PIRTs, including [Boyack] and [Wilson-1998]. Reports that specifically document the development of PIRTs are especially valuable. We have included five of these in the bibliography for this report ([Hanson], [Kroeger], [Rohatgi], [Shaw], [Wilson-1997]). The reader is encouraged to examine one or more of these documents to better understand the intricacies of PIRT development.

4.3 PIRT Construction for V&V

Figure 4 presents yet another modification of a diagram that appears in [Wilson-1998]. The purpose of Figure 4 is to demonstrate that development of a PIRT is a process as well as a product. Emphasizing process gives us valuable insight into key issues of PIRT development and the role of the PIRT as a natural bridge between stockpile requirements and V&V activities.

Figure 4 illustrates the flow of information required to generate a useful PIRT. Questions answered by the various states of this process are:

- What is the stockpile driver for which the PIRT is being developed?
- What are the objectives of the M&S in this context?
- What are specific scenarios that specify the application of the code? (The temporal domain that scenarios may involve can also be important to the structure of the PIRT.)

- What physical phenomena must be implemented via mathematical algorithms and software to allow these code applications? (This question is aimed at identifying as many of the required phenomena as possible.)

- What experimental data are available to support the phenomena identification, for phenomena calibration, and for validating the code implementation of the phenomena?

- What is the importance ranking of the phenomena? What is the method (or methods) used to determine this ranking?

- What are the accuracy requirements for the models of these phenomena?

- What is the current capability of the code to model these phenomena with confidence?

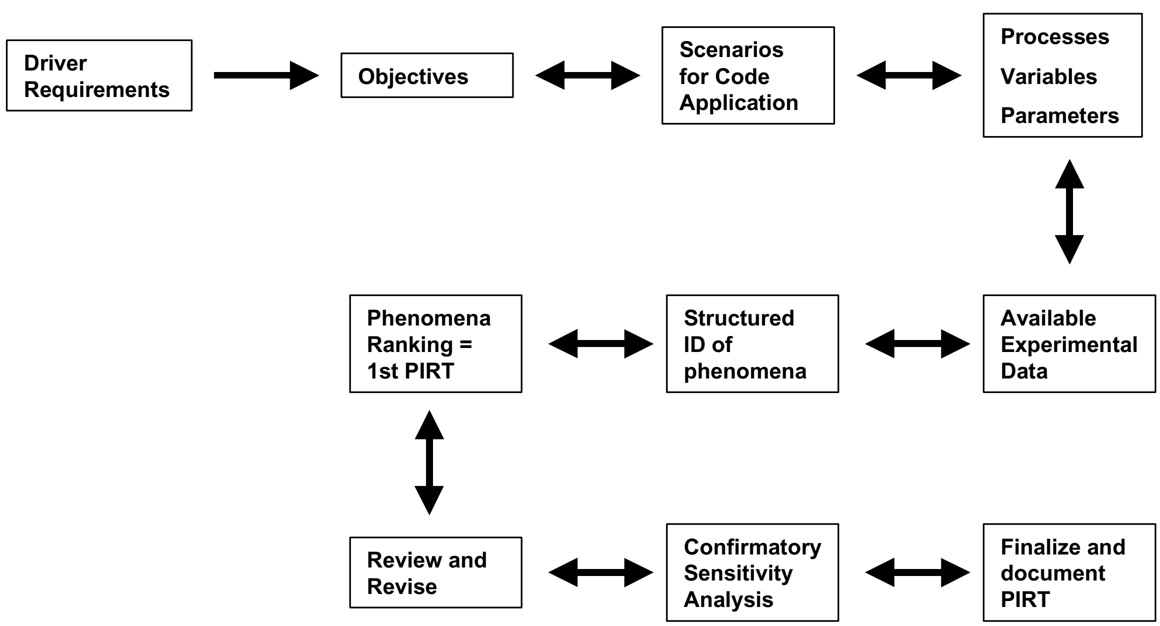

Figure 4. The PIRT process.

In addition, Figure 4 emphasizes that PIRT development is an ongoing process. The process must allow for structured revision of the PIRT. The initial prioritization of phenomena in a PIRT is dependent upon the quantitative characterizations of the M&S requirements that result from the stockpile requirements. These characterizations may vary as the code evolves. In addition, the importance rankings of the PIRT should be confirmed by formal sensitivity analyses. It is unlikely that these sensitivity analyses can be fully performed at early stages of the PIRT process. The importance rankings may also vary as accumulated V&V activities reveal incorrect features in the initial importance rankings. Finally, as V&V activities proceed phenomena that were entirely missed in the initial identification stage may be discovered. When this happens they must be inserted in a revised PIRT.

There is some cautionary guidance available for executing the process outlined in Figure 4. First, PIRTs must be developed using teams that consist of DP personnel, code developers, analysts, and subject matter experts, including experimenters. Second, proper choice of the stockpile driver is critical to the successful development of a PIRT. Too generic a driver will yield a PIRT that is too generic. The importance of phenomena will then be imprecisely characterized. If the driver is too specific, on the other hand, the resulting PIRT will have most or all of the phenomena of great importance. A PIRT that fails to adequate differentiate the importance of phenomena will be relatively useless in guiding V&V activities that are resource constrained.

Use of the PIRT process gives substance to the need to prioritize V&V activities. Since all resources devoted to V&V are limited in realistic code projects, it is imperative to focus effort on those phenomena in the M&S that have the greatest importance. The PIRT is a good tool for choosing these priorities.

Documenting the PIRT must include documenting the PIRT process. It also includes documenting the method used for quantifying the rankings in the PIRT. The specific example discussed in [Wilson-1998] uses a formal logic framework called the Analytical Hierarchy Process [Saaty] to do this. Wilson also recommends a quantitative ranking scale of 1 to 3 or 1 to 5. But a scale of 1 to 9 is applied in [Hanson].

Clearly the choice of methodology and scoring for ranking phenomena in a PIRT is dependent on the stockpile driver, the code, and the personnel that execute the PIRT process. The V&V plan should describe the chosen methodology and scoring.

Let us now consider several generic examples of a PIRT. The references should be consulted for more detailed examples. The important fact to emphasize is that there are many possible forms of a PIRT. It is important that specific code projects use a form that is compatible with their stockpile requirements as well as their codes.

The first example is given in Figure 5. There, a generic form of PIRT has been presented. The underlying assumption is that there is one stockpile driver (not stated in the table), and that two phenomena must be modeled to successfully apply the code to the driver. Two columns give hypothetical answers to questions of (1) how important the specific phenomenon is to the modeling goals and (2) an assessment of how adequate the current status of modeling of this phenomenon is. For example, phenomenon #2 could be in a research status, hence unsatisfactory for stockpile applications, while phenomenon #1 could be modeled by a piece of software that can be purchased. We have also suggested one approach for performing the ranking and prioritization – via purely qualitative measures. How appropriate this is must be a decision a particular code project makes.

Phenomenon	Importance	Current Adequacy
#1	Less	Adequate
#2	More	Inadequate

Figure 5. A Phenomena Identification and Ranking Table (PIRT) – the simplest case.

Another, incrementally more complex example, is given in Figure 6. The general features of this PIRT are similar to Figure 5. We have simply demonstrated that we expect that for ASCI codes, the summary of phenomena will be more complex than suggested by the PIRT in Figure 5. We have continued to use qualitative ranking and prioritization in columns two and three of the PIRT in Figure 6.

Phenomenon	Importance	Current Adequacy
#1	Less	Adequate
#1-A	Less	Adequate
#1-B	More	Inadequate
#2	More	Inadequate
#2-A	More	Adequate
#2-B	More	Inadequate

Figure 6. A Phenomena Identification and Ranking Table (PIRT) – a slightly more complex case.

Figure 7 shows an example of a PIRT that is the same as that in Figure 6, except that the rankings are now quantitative. Both importance and current adequacy are ranked on a hypothetical quantitative scale from 1 to 4, with 4 "highest" for column two, "Excellent" for column three, and 1 "lowest" for column two, "unusable" for column three. The ability to generate a meaningful numerical ranking of this type is obviously highly dependent upon the stockpile driver, the nature of the code project, and the intended evolution of that project.

Phenomenon	Importance	Current Adequacy
#1	2	3
#1-A	1	4
#1-B	2	3
#2	3	4
#2-A	3	3
#2-B	4	2

Figure 7. A Phenomena Identification and Ranking Table (PIRT) – a slightly more complex case with quantitative rankings.

Figure 8 extends the PIRT example in Figure 7 to a case where multiple scenarios of application of the code are represented in the PIRT. How might this happen in reality? To illustrate, in the case of ALEGRA-EMMA, there are two major applications of the code intended for studying NG power supplies. One (say "Use A") is to apply the code to power supply design studies. The other (say "Use C") is to apply the code in qualification activities for future power supply designs. Each application or use of ALEGRA-EMMA relies on the same phenomena for the modeling and simulation. Why do we prioritize each independently? Because the stockpile stakeholder(s) in this case have informed the code team that the accuracy requirements are significantly different between the two uses. In such a case, we easily see that each use might rank phenomena both by importance and current implementation capability differently. For one thing, a phenomenon which might be suitably currently implemented for design, might be inadequately implemented for a certification study. On the other hand, a phenomenon might not be important to achieve the accuracy required for design, but could be a very important for certification. In fact, a PIRT similar to that in Figure 8 has been developed for the ALEGRA-EMMA V&V plan.

Phenomenon	Importance			Current Adequacy		
	Use A	Use B	Use C	Use A	Use B	Use C
#1	2	3	4	3	3	2
#1-A	1	3	3	4	3	2
#1-B	2	4	4	3	2	1
#2	3	3	4	4	3	1
#2-A	3	3	4	3	2	1
#2-B	4	3	4	2	3	2

Figure 8. A Phenomena Identification and Ranking Table (PIRT) – a slightly more complex case with more quantitative ranking approach and more than one way of using the code for the stockpile requirements.

Figure 9 is an illustration of a subset of a PIRT under development for the PEGASUS NG neutron tube modeling code. In this example, the stockpile driver is to predict the performance of NG neutron tubes of a specific type. We observe that this stockpile driver also includes more than the specific application for which PEGASUS is being developed. For example, included under this stockpile driver is the need to model the NG power supply, a stockpile driver for the ALEGRA-EMMA code.

In Figure 9, the stockpile driver has been refined into two broad requirements on the code, similar to our discussion of Figure 8: (1) to quantitatively predict neutron production and (2) to qualitatively predict trends resulting from design changes. These more specific stockpile requirements then determine the phenomena that are required in the PEGASUS modeling. In the case of Figure 9, these phenomena are identified in column one. We discern from study of Figure 9 that the phenomena have been refined to have three major categories (similar to the very crude example in Figure 6). The highest level is illustrated by only one line in this subset of the original PIRT - "PLASMA CUP". The identifiers C-1 ("Inflow Description"), C-2, and so on designate an intermediate level. The lowest level is designated by C-2-a) "Secondary Electron Emission", for example.

Stockpile Driver

Refinement of stockpile driver (code output)

Phenomena

Neutron Tube Simulation	Importance to NG Tube Performance Analysis		Model Adequacy (physics models and/or coefficients)
(Phenomena Modeled)	Predict Neutron Production	Trends with Design Changes	
Plasma Cup (C)			
C-1. Inflow Description	High	High	Inadequate
C-2. Surface Interactions a) Secondary Electron Emission b) Neutralization c) Surface Reactions d) Thermal/Angular Reflection e) Surface Charging	 Low Medium Low Low Medium	 Medium Medium Low Low Low	Inadequate (3D) Inadequate (C) Adequate Inadequate (C) Adequate Inadequate (C)
C-3. Neutral, Ion and Electron Densities and Energy Distributions	High	High	Adequate (2D) Inadequate (3D)
C-3. Gas Phase Reactions: a) Electron Impact b) Excitation Reactions c) Coulomb Interactions d) Charge Exchange Interactions	 High High Medium Medium	 High High Medium Medium	 Inadequate (3D, FK) Inadequate (3D, FK) Adequate Adequate
C-5. Electric Fields			
C-6. Magnetic Fields			
C-7. Sheath			
C-8. Outflow Description			
C – transport or interaction parameters, FK – full kinetic simulation			

Figure 9. A partial Phenomena Identification and Ranking Table (PIRT) for a complex stockpile application involving a NG neutron tube [Johannes].

The three dimensions we seek in a PIRT are clearly identified in the example in Figure 9. The associated physical phenomena are identified, their importance to the two major code requirements is ranked in the middle two columns, and an assessment of the current status for the phenomena model formulation and implementation is given in the right column.

As one final example, we illustrate how a PIRT can provide some conceptual guidance for the development of verification test plans. In Figure 10 we illustrate the progression from stockpile requirements to the identification and ranking of phenomena in a PIRT. We then illustrate a subsequent step, in which a series of verification test problems are developed to test the various phenomena identified by the PIRT process. Some of the problems selected may overlap several phenomena, while others may apply only to specific phenomena. The PIRT guides emphasis for these choices of test problems through the ranking of the phenomena. Tests that are aimed at the most important phenomena clearly should command more attention than those that are not. Additionally, although not explicitly depicted in this figure, the PIRT provides information that will help establish the assessment metrics associated with these problems. This entire issue is discussed further in Section 6.

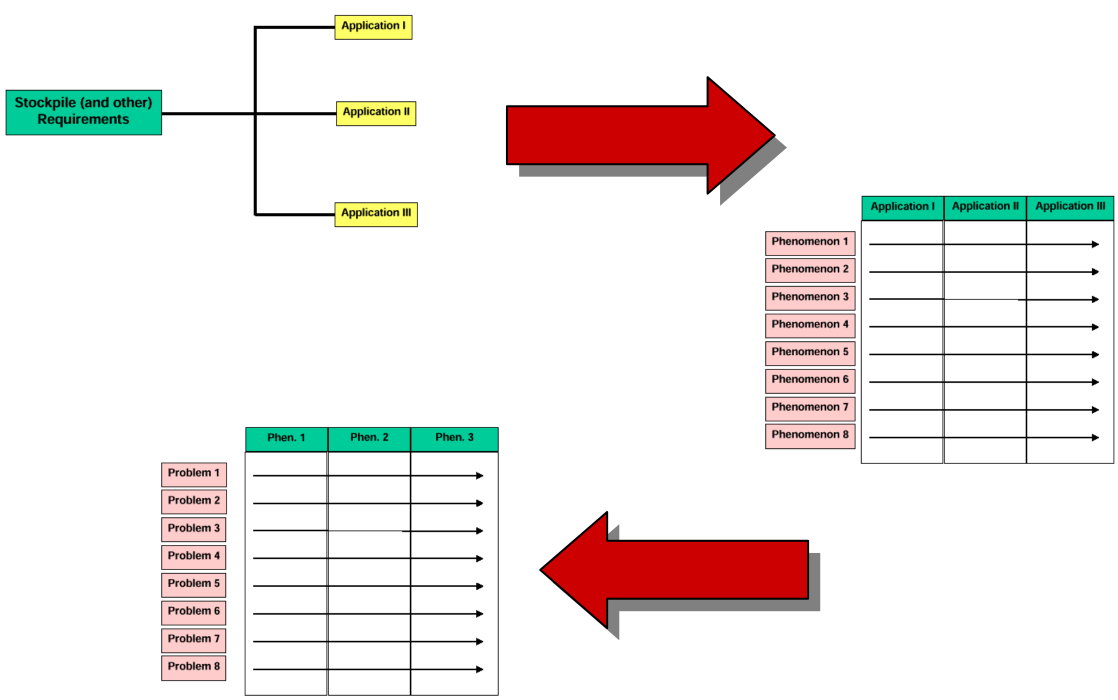

Figure 10. Application of the PIRT to defining verification test plans.

In summary, Figure 4 captures the process that is used in developing a PIRT that supports V&V activities. The V&V plan should describe the PIRT process in enough detail to portray where this information is used to produce the PIRT defined in the plan.

4.4 Conclusion

V&V plans must contain a PIRT for the targeted stockpile driver. The PIRT is an essential bridge between the stockpile driver and detailed V&V activities, as suggested in Figure 2.

We do not provide rigid or explicit guidance for the development and documentation of a PIRT. The development of the PIRT depends on how elaborate the application of the code is for the relevant stockpile requirements. Whether a quantitative ranking or only a qualitative ranking of phenomena is required depends greatly on what the DP stakeholders tell the code project about the role of the code for the stockpile driver.

The PIRT is obviously critical for validation activities, because the modeling status, importance and completeness of the phenomena identified in the PIRT are fundamental information needed to focus the code's validation effort. Yet, the PIRT is also an important guiding principle for code verification and, ultimately, for uncertainty estimation. The priorities of the PIRT clearly serve to focus verification activities, particularly the verification testing activities discussed later. A verification test plan should ideally mirror the priorities given in the PIRT. Discussion of the logic behind verification test plans for the code (see Section 6) should have relation to the priorities established by the PIRT.

From a programmatic view the most important part of our guidelines has now been defined. We have emphasized the need for an accurate presentation of stockpile requirements and the culmination in the development of a PIRT. We reiterate that there are two main reasons for attaching so much importance to the exposition of requirements. First, they insure contact of our V&V efforts with DP at Sandia. Second, our chosen model for the actual practice of verification and validation is "requirements based V&V." This model emphasizes that code V&V activities follow logically from code requirements. We now turn our attention to guidelines for both verification and validation test plans.

4.5 Content Criteria Summary

The following summarizes the key content criteria reflecting the V&V plan elements discussed in this section.

PIRT1:	The PIRT is present in the V&V plan.
PIRT2:	The PIRT process methodology is described.
PIRT3:	Phenomena in the PIRT are ranked by a rational scoring system.
PIRT4:	Current M&S status (capability) for each of the phenomena in the PIRT is presented.

5. Software Quality Engineering (SQE)

5.1 Introduction

SQE guidance for ASCI codes is now the subject of DOE ASCI Program SQE Goals, Principles, and Guidelines [DOE-2000b]. SQE is an important contributor to establishing evidence of required M&S capability. It is essential to understand the role that the code team assigns SQE in its V&V plan. The criteria we define for SQE content in the V&V plans are drawn from the DOE guidance.

The relationship of goals, principles, guidelines, and practices is illustrated in Figure 11, which is extracted from [DOE-2000b].

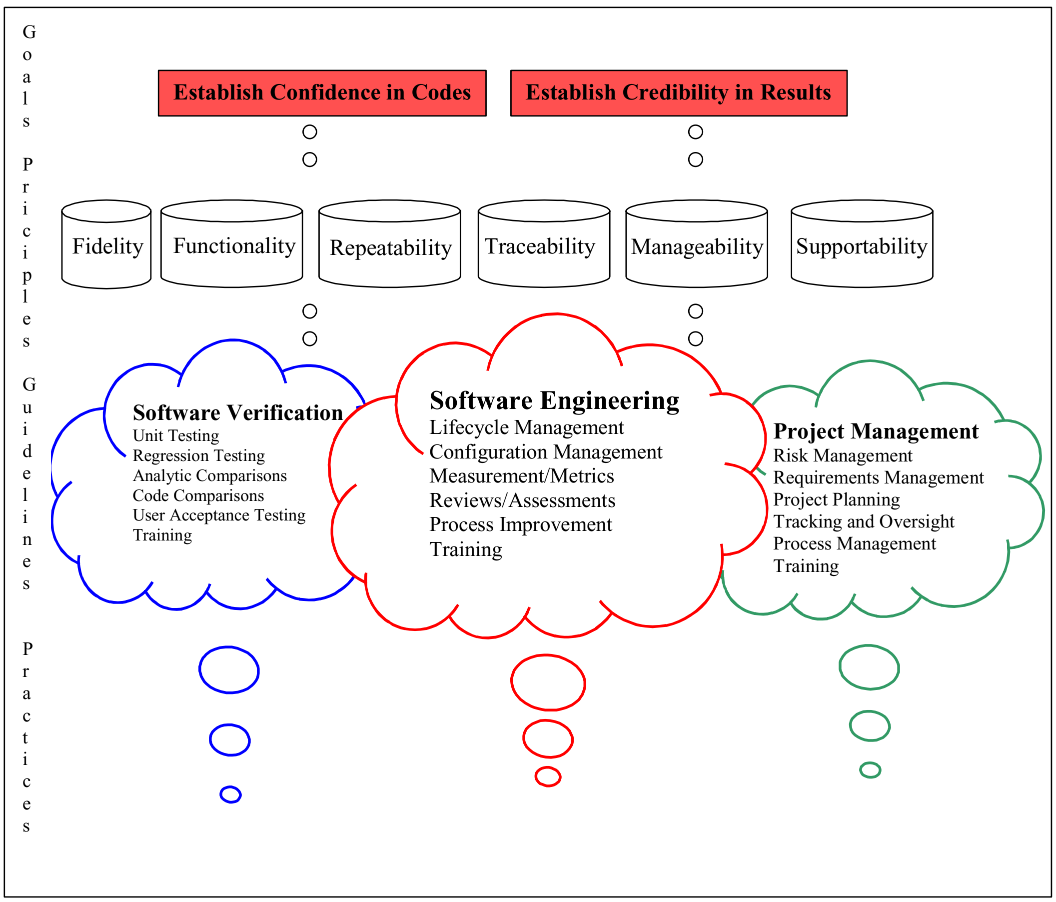

Figure 11. The relationship of SQE goals, principles, guidelines, and practices [DOE-2000b]

The selection and implementation of practices by an organization or project should support the goals of the ASCI program. There are a few principles that if adhered to would likely achieve the goals or at least directly support the achievement of the goals. Guidelines for SQE activities have been derived so that when the activities' attributes satisfy the principles in some measurable way, the goals will be directly supported. The specific SQE practices should be adopted by the organization or project in a manner so as to implement the guidelines in accordance with an appropriate tailoring strategy.

The SQE practices that are essential to implementation of the DOE guidelines are defined by individual laboratory specification. The practices that are specific to the Sandia implementation are documented in [Hodges].

5.2 SQE Guidance

The V&V plan should summarize content in the policy areas described below. The use of the documentation tree approach described in Section 2 above is highly recommended in assembling evidence related to the guidelines defined here.

Guidelines to support the principles are specified below in terms of expected activities. These guidelines for expected activities are in the general areas of verification and validation, software engineering, and project management. Specific methods, techniques, and tools are used to implement practices. Practices and associated methods, techniques, and tools are assumed to be site-specific, and implemented according to the ASCI program and development team priorities. The guidelines for specific process elements are explained in greater detail below.

5.2.1 Verification and Validation

Verification and Validation is the determination that requirements are accurately and correctly implemented, and that requirements are adequate from the perspective of the intended uses of the software. Validation activities should include user acceptance testing, documentation of results, and capabilities definition and management. For physics and engineering codes, validation activities should include uncertainty analysis and testing against real world data. Verification activities should include testing against analytical data and other software application output. General V&V activities should include technical reviews and code-to-code comparison of results. These activities set the stage for establishing the confidence in the use of the code within an application-specific validation scenario, and comparison of results to experimental data. Although the Figure 11 from [DOE-2000b] primarily covers software verification, Sandia includes the full spectrum of software verification and validation as an integrated part of the V&V approach.

Software Verification and Validation Practices - Software verification activities include :

Technical Reviews—the activity of evaluating the technical soundness of work products. This includes analyses to find mismatches or faults between the specification and the design, code, or documentation.

Unit Testing—the activity of testing code units against their requirements, specifications, and design. This activity involves the development and documentation of unit-test drivers and test-case inputs. This activity requires valid work products to be provided by the software developers that clearly and adequately define the requirements, specifications, and design. Unit testing should be developed and performed by the software developers during the development life cycle. It should be traceable and repeatable by an independent V&V team, where it is appropriate to do so.

Regression Testing—the activity of regularly *building* the code and *executing* a series of tests designed to verify that the code works as expected for all computational platforms supported. Minimally, such testing should be done when either the code or operating platform changes. This activity includes the development and maintenance of a regression test suite. This test suite should be designed to exercise as many of the code features as possible. The regression test suite should include integral and unit tests, as appropriate.

Comparison Techniques—the activity of utilizing additional comparison techniques within the code development team to ensure requirements are being met on a local scale. These activities could include comparing to analytic solutions, and to other codes.

User Acceptance Testing—the activity of determining if the work products satisfy the needs of the intended users. This activity should include evaluation of applicability and usability from the end-users' points of view. It is also intended to help build the users' confidence in the codes and their belief in the credibility of the results.

Training—the activity of developing the skills and knowledge of those individuals responsible for software verification activities.

Software validation activities include comparing results with expected numerical, mathematical, and conceptual solution behavior for specific applications and environments. These activities set the stage for the qualified use of codes within an application-specific scenario, and for comparison of results with experimental data. Content in this latter area is discussed in detail in Section 7.

5.2.2 Software Engineering

Software Engineering is the systematic, disciplined, and quantifiable approach to the development, operation, and support of software, i.e., the application of engineering to software. Activities should include identification of a life cycle model; development (e.g., requirements, design, implementation, test, release); operation (e.g., execution on multiple platforms, regression tests); support (e.g., change analysis, implementation, test, and release); measurement of product and process attributes; reviews and assessments of products and processes; and training on software engineering activities. The balance among activities, and the relationships with modeling and simulation, verification and validation, and project management depends on many factors including the maturity of the software. Expected software engineering practices are summarized below.

Life-Cycle Management—the activity of organizing requirements, design, construction, test, and support activities into a time-based work flow. Many life-cycle models exist that could conduct this activity. The life cycle model selected and the specific activities of requirements, design, construction, test, and support should have well-defined interfaces with the other software engineering support areas, and should be based on the guiding principles that best achieve the intended applications and overall ASCI V&V Program goals.

Requirements activities should include methods for gathering requirements from the scientific application modeling domain; analyzing and documenting models that depict required system data, function, and behavior as allocated and traced throughout the application components; verifying that requirements are met in the application design and implementation; and managing any changes to the requirements.

Design activities should include repeatable methods for translating requirements information and models (scientific and software) into representations that convey software data structure, architecture, algorithms, and interface features.

Construction activities should include methods that implement a specific software solution of the design, and that can be traced to the design and verified to the specified requirements.

Test activities should include methods to verify the software construction from unit to integrated software components to scientific model application design and requirement specifications, where applicable. These activities overlap with software verification activities to the extent that the activities use similar test suites and results to achieve the required confidence in the software implementation.

Support activities should include methods to manage changes to the implementation of requirements, design, construction, and test work products due to defects that are found, enhancements that are needed, or the natural technological evolution within the application domain. Support activities should include effective interfaces with other software engineering activities such as configuration management for controlling the changes to, and updated releases of, work products.

Configuration Management—the activity of identifying the configuration items in a system, controlling the release and change of those items throughout the system's life cycle, recording and reporting the status of the items and associated change requests, and verifying the completeness and correctness of the items. Configuration management activities are organized into version management, issue tracking, and release management. Version management is the identification and control of the versions of all products, both by individual pieces (e.g., software module) and by appropriate groupings (e.g., set of software modules that constitute an executable program). Issue tracking is the identification and tracking of problems and associated corrective actions, proposed changes for enhancements, and the workflow of activities to accomplish implementation of the change. Release management is the control of product promotion, from development to production use.

Measurements/Metrics—the activity of collecting information for the characterization, understanding, and evaluation of processes and products. Metrics should show how selected site-specific practices satisfy related attributes of specified principles and consequently contribute to meeting the V&V program's goals of confidence in codes and credibility in results. Only metrics that can be demonstrated to meet project and/or the V&V program's goals should be chosen.

Reviews/Assessments—the activity of examining and evaluating the quality of a process or product. Reviews/assessments should be conducted on work products from all life-cycle phases to catch defects as early as possible. Formality and scope of reviews/assessments, like other activities, should be tailored. Results of the reviews or assessments should be recorded. There are basically two types of reviews:

1. Management reviews evaluate and communicate status of the project with regard to schedule, cost, and performance; determine whether processes are being followed correctly, particularly with regard to impact on performance, cost, and schedule; and may be internal to the project, or include external personnel and stakeholders.
2. Technical reviews evaluate technical soundness of work products and processes; include analyses to find mismatches or faults between the specification and the design, code, or documentation; and are conducted by relevant domain experts.

Process improvement – the activity of baselining the performance of a process through a documented characterization of the actual results achieved by following the process, determining how the process should be improved in comparison with the actual results, and establishing an approach to achieving the improvement.

Training – the activity of developing the skills and knowledge of those individuals responsible for software engineering activities with respect to relevant procedures, tools, and domain knowledge, as they apply to the ASCI program. It also includes training on the use of the developed software products, as well as their domain, scope, and applicability.

5.2.3 Project Management

Project Management is the systematic approach for balancing the project work to be done, resources required, methods to be used, procedures to be followed, schedules to be met, and the way that a project is organized. Activities should include: identification, analysis, and mitigation of project risks; controlling requirement changes; planning for project tasks, schedule, and cost; tracking project progress and status; providing oversight of process improvement; and training project personnel in management activities.

Risk Management – the activity of identifying, addressing, and mitigating sources of risk before they become threats to successful completion of a project. Risk management elements are:

- risk assessment (identifying, analyzing, and prioritizing);
- risk control (management planning, resolution, and monitoring).

Requirements Management**—**the activity of capturing, tracking, and controlling requirements, as well as any changes to them. This establishes and maintains a common understanding, between customers and development teams, of the requirements to be addressed by the project. This agreement should be the basis for planning and managing the project.

Project Planning**—**the activity of establishing a reasonable plan for performing and managing the project; work products should include, but are not limited to, a statement of work, constraints and goals, project plan, project timeline, an assessment of resources that will be needed, and availability of those resources.

Tracking and Oversight**—**the activity of tracking and reviewing the project accomplishments and results with respect to the project plan, and taking corrective action as necessary based on actual accomplishments and results.

Process Management**—**the activity related to planning, defining, implementing, monitoring, measuring, and improving processes under Project Management; and producing process documentation and improvement plans.

Training**—**the activity of developing the skills and knowledge of those individuals responsible for Project Management activities.

5.3 Content Criteria Summary

The following summarizes the key content criteria reflecting the V&V plan elements discussed in this section.

SQE:	Content required by DOE and Sandia SQE policy documents is present.
SQE1:	Software V&V.
SQE2:	Software Engineering.
SQE3:	Project Management.

(Page Left Blank)

6. Verification Testing

6.1 Introduction

In addition to the validation testing requirements, code verification requirements are also specified by the PIRT. The PIRT emphasizes the implementation needs for the code, along with priorities. Therefore, the PIRT connects the software implementation priorities to the stockpile driver. Focus on mathematical models is required to develop the PIRT. We presume that some kind of assessment of the current capability of the math models underlying the software implementation of the key phenomena listed in the PIRT has been performed to develop priorities. This assessment provides natural focus opportunities for the software verification process.

The ultimate goal of the software verification process is to increase our confidence in the implementations of the required phenomena and their numerical behavior. The V&V plans should address both SQE and the verification testing process. Information and priority guidance from the PIRT tend to intersect each of these areas. However, the PIRT can also aid in designing and applying the verification test plan. This test plan must clearly address verification requirements that are emphasized in the PIRT. For example, it is reasonable to expect that the intensity and degree of effort to be applied in verification testing will correspond to the priorities established in the PIRT. The complexity of verification testing also reflects the complexity of the phenomena detailed in the PIRT. Similarly, the priorities of the PIRT, and coupling of the phenomena listed there, may also provide guidance for where to implement and formalize SQE activities for the code.

The level of detail and the relevance of the software verification requirements will depend on the specific code project and its PIRT, as well as on the anticipated complexity of the verification test process. For purposes of the peer review that the Sandia V&V program is initiating [Pilch], enough detail is required to understand the definition of the verification requirements and their impact on the verification test process.

Three questions need to be answered in order to determine if the software verification requirements are sensible and achieved. ***Why*** was the verification test suite chosen? ***How*** was the verification test suite chosen? ***When*** are verification tests passed? A clearly defined requirement should lead to a need for testing. On the other hand, a test problem taken in isolation should also have an implied software verification requirement associated with it. It may be true in this latter case that this requirement will have to be "reverse engineered" as discussed in Section 4.3.

The verification requirements and their success metrics serve as the most basic instrument that can be applied to assess the progress of the verification process for the code. We emphasize that it is implicit that all V&V plans will be subject to periodic revisions, reflecting inexorable changes in development priorities. The plan should attempt to develop a coherent picture of the requirements that are driving code development available at the time of writing and at those times of periodic revision. It is also appropriate to give definition to the expected revision process. This could be done, for example, by using a formal requirements management framework elaborated in the V&V plan.

6.2 Physics and Engineering Verification Testing

In this section we focus on guidelines associated with the test suite developed to assess the performance of the physics and engineering algorithms implemented in the code. Included in this category is the mathematics that is required to correctly solve the physics and engineering algorithms, for example linear solvers. For short, we will call this test suite the Verification Test Suite (VERTS). The test suite thus under current discussion does not include other more software oriented testing such as that which naturally occurs in SQE practices [Hodges]. This view does not imply that the two are logically disjoint in some deep fashion from the overall problem of code verification.

A key issue is the great weight that has traditionally been assigned to proper execution of problems in a VERTS. In fact, this type of test suite is one of the critical acceptance factors in the exploratory code development process discussed by Ambrosiano and Peterson [Ambrosiano]. There is no evidence that suggests that this weight has changed qualitatively for the ASCI codes being developed at Sandia. Acceptable performance of a computational science code on a VERTS is the main factor that determines whether the code is ready for validation studies or not, both in the minds of the code team and in the opinion of potential users. It is extremely important to properly emphasize such a test suite in a rational V&V plan. It is not the position of these guidelines that the definition of the VERTS for a code, and successful execution of the VERTS by the code, constitutes complete "verification" of the code. Rather, the VERTS is one of several factors, albeit an very important one, that address the ultimate verification of the code for its application to the stated stockpile driver.

In the following discussion, it is useful to introduce three dimensions that address the three questions posed in Section 6.1 above. These dimensions measure independent categories of knowledge of the VERTS. The first of these dimensions is the ***structure*** of the VERTS, or the logical principles underlying it. The second dimension is the ***construction*** of the test suite, or the specific means chosen by the code team for populating the VERTS. The third dimension is that of ***assessment*** versus the VERTS, specifically the criteria that are applied for deciding whether or not the code has passed or failed a given test. In requirements driven V&V the PIRT provides constraints and connections for the structure of the VERTS, as well as a filter on the choice of problems and criteria defining code acceptance *vis a vis* its execution of the VERTS. The PIRT is germane to the VERTS. It is important to discuss the factors that relate the VERTS to the PIRT in the V&V plan.

Complete documentation of a VERTS in terms of these three dimensions is probably well beyond the scope of the overall V&V plan that is the focus of the present guidelines. Such documentation will be quite technically detailed and probably quite long. A useful device, therefore, is to reference an independent document describing the VERTS, perhaps via an executive summary. This approach maps nicely into the conceptual documentation tree suggested in Figure 1. How this is decided is up to the individual code team.

We discuss each of the three knowledge dimensions in the following paragraphs.

6.2.1 Structure of the VERTS

The ***structure*** of the VERTS is mainly the logic by which this test suite is defined and applied. It answers the question of ***why*** the VERTS was chosen. It is the first information that must be understood when considering the potential for success of the VERTS upon the ultimate code verification challenge. In the planning approach advocated by these guidelines, structure for the VERTS should reflect the PIRT. The structure of the PIRT is therefore a good candidate for structuring the VERTS.

In Version 1 of these guidelines [Trucano], VERTS structure was defined through the use of three Tiers, or categories, of test problems:

> **Tier I** - tests with exact analytical solutions;
>
> **Tier II** – tests with semi-analytic (reduction to quadrature, to simple ordinary differential equations, etc) solutions;
>
> **Tier III** – idealized problems suitable for code comparison exercises.

This structure for verification testing is advocated in the AIAA V&V Guide [AIAA]. It reflects the critical importance of accurate assessment of coded execution of verification test problems. While this approach to structuring the VERTS is adequate for V&V planning (and commonly used in previous code projects) it does not directly capture the structure of any PIRT. It would be difficult to even reverse engineer the connection of the VERTS to the PIRT for this structure.

The structure defined for validation test problems in Section 7 is a more appropriate and attractive choice for structuring the VERTS. That structure is specifically intended to reflect the definition of the PIRT, in particular increased complexity and coupling of phenomena and their models. This structure applied to the VERTS is:

> **Tier I** – tests of individual algorithms (or more generally physics/engineering code modules);
>
> **Tier II** – tests of coupled algorithms (more generally physics/engineering code modules);
>
> **Tier III** – tests of integral algorithm capability (more generally integral physics/engineering implementation).

The fourth category of validation problem defined in Section 7 is Tier IV, a category intended for application qualification of the code. This category appears to apply uniquely to the validation test component. We do not advocate a corresponding classification for the VERTS, although this is ultimately a decision for the code team to make.

For example, consider a radiation-hydrodynamics application of the ALEGRA code. To map to the structure defined above, the Tier I test should then be chosen to separately test ALEGRA algorithms designed to calculate the propagation of shock waves, the compressible flow of materials, and the flow of radiation, as well as other issues such as material model performance. The individual algorithms being tested by these problems will be easily mapped to the PIRT as the numerical solution implementations of individual phenomena specified in the PIRT.

Tier II problems would then be designed to test coupled radiation-hydrodynamics calculations, but in simple (or simpler) cases than required for the ultimate application of the code. Simplification could be specified by geometric simplicity, or by reductions in the amount of coupled physics required (such as the use of a simple analytic opacity rather than a realistic opacity). The couplings should reflect coupled phenomena expressed in the PIRT.

Tier III would define tests of coupled capability similar to the requirements of the needed application. This assumes that such tests can be defined, of course. The level of difficulty of defining test problems as the module coupling increases may itself increase dramatically. Such a test may require a code comparison exercise between ALEGRA and another existing code for a problem that fully expresses all of the complexity of the application the PIRT has been designed for, such as a realistic geometry.

The advantage of the type of structuring of the VERTS that is defined above is that it can be enforced to reflect the prioritization of the PIRT. In turn, it also replicates the recommended structure for validation testing, except for the Tier IV qualification activity. This structure serves as a guide for the development of a VERTS that should emphasize the most important phenomena expressed in the PIRT. There is an advantage to this approach, but ultimately the choice of VERTS structure and the logic it implies is still the prerogative of the code team. If assessment is the primary concern of the code team then the original Tier structure can be applied. (Note that the original Tiers could also be used to provide a secondary structure within each of our newly recommended Tiers.) Whatever structure and logic are chosen, they should be documented clearly in the V&V plan.

6.2.2 Construction of the VERTS

Discussion of the ***construction*** of the VERTS answers the basic question of ***how*** the VERTS is actually populated with appropriate test problems. This information must convey how the code team selects those tests that successfully express the desired structure of the VERTS.

There are several possible approaches for constructing the VERTS. First, individual test problems, or test suites, representing "***industry standards***" may be available. If they exist, such test problems clearly are important and should be used. The code team would likely have to defend in peer review a decision to *not* use such problems. The likelihood of existence of a set of industry standards that define test problems for even part of the VERTS is rather small, however. A rather complete set of such problems has been traditionally used by Los Alamos National Laboratory and Lawrence Livermore National Laboratory for their weapons codes. But, these laboratories are in the unique position of being able to de facto define their own industry standards through this practice. At the far end of the spectrum, but still constituting what we mean by an industry standard, are individual test problems that appear again and again in verification activities for a specific engineering or physics discipline. For example, the shock hydrodynamics test problems quoted by Woodward and Collela [Woodward] constitute industry standards in the sense we mean because of their widespread use in assessing numerical performance of shock hydrodynamics algorithms.

Because of the breadth of disciplines spanned by the ASCI codes under development at Sandia it may be that particular codes have standard test problems in this sense available to them. If this is the case and the problems are used to construct part of the VERTS for the code this should be made clear in the detailed documentation of the VERTS, as well as the specific source of the problems.

In the absence of some kind of clear standard for test problems, a second approach to populating part or all of the VERTS for a particular code is ***consensus***. The appropriate community for establishing this consensus is the code team and their actual and potential user community. In particular, expert opinion regarding discipline specific important test problems will be available to those codes participating in this program. The consensus approach takes advantage of an otherwise more restrictive approach, simple expert opinion, by including the larger formal community attached to the specific code. The community that establishes test problem consensus, of course, could simply be the internal Sandia community associated with the code, the code team and their internal users. It could also include external consultants associated with the code.

Consensus is embedded in any industry standard set of test problems. It is then a simple matter to select those problems as part of the VERTS for a code. Developing consensus is the hard part of the job. If a Sandia code is sufficiently unique in its technical discipline, the consensus process will likely be a step toward establishing "industry standards" for test problems in that subject. By this result, the effort of the code team in establishing such a component of the VERTS will have impact beyond the Sandia ASCI community. Consensus development of test problems for the VERTS therefore also addresses the need to withstand external scrutiny. That alone is a good reason for adequately documenting the consensus process and its results.

Further opportunities to understand and build on the use of industry standards (if available) and a consensus approach present themselves in the peer review process Sandia is establishing for the V&V program [Pilch]. One of the focus areas for the Level 2 peer review is assessment of the VERTS. Such a peer review could develop evidence supporting the construction of part of the VERTS using consensus or reveal additional issues that may have been neglected.

A third approach to constructing elements of the VERTS is to simply construct specialized test problems defined to address specific needs arising in the structure of the VERTS. This approach will be called ***technological*** in this document, mainly because technology of one kind or another is expected to influence its success. Two examples are worth considering. In the first case, a code team may elect to populate at least part of the VERTS based on issues of coverage of key code modules or paths or both. The issue of regression testing, for example, might drive such needs. In this case, the construction of these tests will likely rely on coverage analysis technology, such as [PureCoverage]. An alternate approach to custom building selected verification test problems is the use of manufactured solutions [Salari].

We will conclude this section by emphasizing the need for the V&V plan to describe the process used to construct the VERTS. Whether there are existing industry standards, whether a VERTS is developed by the consensus of a code team and complementary expert community, or whether a substantial percentage of the VERTS is developed piecewise by accommodating special needs through the use of technology, the V&V plan should describe the approach chosen and its results. Full details will probably be beyond the scope of the V&V plan itself, but they should be available in additional documentation devoted to the verification test suite.

6.2.3 Assessment of the VERTS

The Tiers for the VERTS suggested in the original version of these guidelines and mentioned above are structured to address the following questions. ***When*** is it decided that a given test in the VERTS is passed or failed by the code? Is the assessment as simple as pass/fail or are there intermediate possibilities? The original Tiers were designed to move from very simple determinations of passing (failing) a test problem (Tier 1) to very complex determinations (Tier 3). In the present version of the guidelines we have elected to present the issue of ***assessment*** as separated from the question of how to structure the VERTS. It is important to verify the code to the full complexity of coupled models to the greatest extent possible.

The basic issue is to provide a set of metrics that determine whether or not the code has "passed the test." Definition of success is highly specific to the subject matter discipline, the code, and the specific test under consideration. These guidelines will not attempt to discuss what is meant by "success." This is strictly a matter for the code team and their requirements for applying the code. However, there is a clear philosophy underlying the establishment of specific success metrics for each problem in the VERTS. There is simply no better practical statement of the readiness of a code for validation studies than its accumulated success in executing the VERTS. This success should be spelled out as clearly as possible, hence the need for clear, hopefully quantitative, VERTS success metrics. Obviously success metrics, whether explicit or implicit, carry the measured verification status of the code with them. Success metrics are clearly linked to the PIRT. In the ideal case, success metrics will be directly determined by elements of the PIRT. For example, if a certain phenomenon needs to be calculated to a given accuracy for the canonical application of the code to the stockpile driver, this places constraints on how accurately algorithms implementing that phenomenon must perform when compared with test problems.

It is important for the V&V plan (or the associated VERTS documentation) to define success metrics as clearly as possible. The result of passing test problems in the VERTS in the sense of the identified metrics is that testing efforts should go on to emphasize other problems. Failure to pass tests as measured by stated metrics focuses additional effort (assuming that the structure of the VERTS has been properly defined to begin with) to better execute the tests. Defining acceptance metrics goes a long way to answering the following question: Why waste further effort on tests that are being executed within an acceptance tolerance?

Metrics will vary from being extremely quantitative – energy is conserved to 10^{-6} %; iterations are carried to machine precision; the difference between the accepted result of a test and a code calculation is smaller than a stated threshold – to completely qualitative, such as a "view graph norm" comparison with a different code calculation. It is the code team that makes these decisions and defines the appropriate metrics. The V&V plan should directly state the acceptance metrics for the VERTS, or these metrics should be defined in separate VERTS documentation. These metrics will be developed as an agreement among the code team, their user community, and the larger expert community associated with the subject matter area of the code.

6.3 Content Criteria Summary

The following summarizes the key content criteria reflecting the V&V plan elements discussed in this section.

VERTS:	A Verification Test Suite has been constructed and documented.
VERTS1:	The structure and logic of the VERTS is addressed.
VERTS2:	The construction of the VERTS is addressed.
VERTS3:	The acceptance metrics for the VERTS are addressed.

7. Software Validation Requirements and Validation Plan

7.1 Validation Requirements

The PIRT is the most important connection between key stockpile requirements and key physical phenomena for the M&S to be implemented in the software. The most important task in the validation plan is to logically connect the required validation activities and their priorities to the elements of the PIRT. The V&V plan should carefully document this linkage. By doing this, the V&V plan also demonstrates that the process of assessing the fidelity of the implemented models in the evolving code is responsive to the stockpile requirements that have been used to develop the PIRT.

The following information is required for suitably documenting the validation activities:

- Establish connections between the Validation Test Suite (VALTS) and the PIRT elements.

- Describe the experimental data requirements associated with the validation activities.

- Describe the dependence (known or hypothesized) of assessment of the fidelity of the code physical models upon experimental data.

- Describe how the assumptions underlying the implemented physical models may or may not be subject to experimental investigation.

- State the dependence of the code upon experimentally measured quantities; describe critical experimental facilities or capabilities that must be applied to generate these quantities; state the role of calibration for using these data in the ultimate accuracy of the software.

7.2 Validation Test Suite (VALTS)

7.2.1 General Requirements

The VALTS will contain the following elements:

- A definition of the four-tiered VALTS described below. The discussion should include the required quality of experimental data for executing the elements of the validation test suite. The definition should be compatible with the additional description of the validation test suite content provided below.

- A description of existing experimental data that satisfies some or all of the requirements of the VALTS. As emphasized in Section 2, full detail need not be provided in the V&V plan. Use of additional documents in the documentation tree is encouraged.

- A description of new experimental activities required to support code validation activities if existing data are insufficient. This description should include needed experimental facilities, requirements for specialized diagnostics and test equipment, and estimated personnel needs. Prioritization of new experimental work should match the priorities established by the PIRT.

- A discussion of supplementary technologies that are used or would be useful for validation. For example, if uncertainty quantification is to be applied as part of the validation activities, describe its planned use. How are these technologies implemented or what implementation is planned? What are barriers to their use?

7.2.2 Structure, Construction, and Assessment of the VALTS

As for the VERTS discussed in Section 6, in discussing the VALTS it is important to introduce three dimensions, or independent categories of content. The first dimension is ***structure***, or the logical principles underlying the formation of the validation test plan. The second dimension is the ***construction*** of the validation test suite, or the specific means chosen by the code team for populating the validation test suite. The third dimension is that of ***assessment*** versus the validation test suite, specifically the criteria that are applied for deciding whether the code validation was sufficiently successful. In requirements driven V&V, the PIRT provides constraints, connections, and priorities for the structure of the VALTS, as well as criteria defining code acceptance *vis a vis* its execution of the VALTS. The PIRT is particularly important in defining acceptance criteria for the problems in the VALTS. These criteria should reflect the importance weighting and priorities originating in stockpile requirements as defined by the PIRT. The connections between the PIRT and the structure, construction, and assessment criteria defined for the validation test suite should be explicit.

The detailed comments about structure, construction, and assessment for the VERTS given in Sections 6.2.1 through 6.2.3 carry over almost intact to discussion of the VALTS. We will not reproduce those comments here. We suggest that the reader revisit these sections at this point to recall that guidance.

One essential difference between the VERTS and the VALTS appears in the discussion of structure. The structure we require for the VALTS consists of four tiers. These are defined below.

Tier I – Assess the degree of accuracy to which separable effects (or single phenomena) in the code correctly represent the real world.

Tier II – Assess the degree of accuracy to which coupled effects between distinctly identified phenomena in the code correctly represent the real world.

Tier III – Assess the degree of accuracy to which integral phenomena in the code, in which many coupled effects may be present, correctly represent the real world.

Tier IV – A "Qualification Experimental Campaign" or confirmatory experimental activity designed to assess the readiness of the code for stockpile computing and application to stockpile problems associated with the chosen driver. It is essential that this validation activity be performed in conjunction with the Sandia weapon design community and in coordination with additional experimental opportunities that may arise in the normal course of nuclear weapon program work at Sandia.

Tiers I through III reflect the AIAA validation test hierarchy [AIAA] of Unit Problems, Benchmark Problems, Subsystem Problems and Full System Problems. Unit and Benchmark problems are mainly contained in our Tier I specification. Our Tier IV has no analog in the AIAA classification.

Tiers I through III should emphasize the phenomena and their couplings that are elements of the PIRT. Tier IV problems are not clearly expressed by the PIRT in all likelihood. Rather, this is a termination activity aimed at qualification of a frozen code for the stockpile driver. Further discussion of this issue is given in Section 8.

Executing the VALTS structure is marked by increases in the complexity of associated verification and validation activities, as well as the experiments required to support the validation activity, as one progresses through the Tiers. As confidence is gained at lower complexity (Tier I), the desired complexity of the validation activity should increase. Progressing through the Tiers is a continual process that seeks to establish ultimate confidence in the fidelity of the code to represent the complexity and breadth of the physical phenomena associated with the stockpile driver. Culminating the process, one would expect carefully instrumented full-system hardware/model confirmatory experiments to be conducted synergistically with the code project (Tier IV). Movement through the tiers also intrinsically focuses experimental components of validation testing.

The logic by which the code project expects to assess the VALTS should be presented in detail in the V&V plans. Here, for example, is one important application of the PIRT analysis. PIRTs can be used to order the effort expended on first separable, then coupled phenomena, to best achieve the final integral capability. A defined qualification activity can also be used to organize a validation effort aimed at ascending these tiers. Code projects should be able to develop evidence of their ability to execute the tiers, or to identify roadblocks that prevent them from doing this, such as missing experiments.

7.3 Content Criteria Summary

The following summarizes the key content criteria reflecting the V&V plan elements discussed in this section.

VAL1:	The relation between validation activities and the PIRT is defined.
VAL2:	The data requirements associated with the validation activity are described.
VAL3:	The dependence of assessment of the fidelity of the code physical models upon experimental data is described.
VAL4:	The opportunities and obligation for experimental investigation of the assumptions underlying the implemented physical models are discussed.
VAL5:	The dependence of the code models upon experimentally measured quantities is discussed.
VAL6:	The prioritized experimental needs are described.
VAL7:	The anticipated use of technologies like uncertainty quantification in the validation activities is described.
VALTS:	A Validation Test Suite is constructed and documented.
VALTS1:	The structure of the VALTS is described. The structure is compatible with the PIRT and the four-tiered approach described in these guidelines.
VALTS2:	The construction of the VALTS is addressed.
VALTS3:	The acceptance metrics for the VALTS are addressed.

8. Guidance for Stockpile Computing

8.1 Code Qualification

There must be means available for defining termination points for some aspects of V&V. There is insufficient time, money, and human resources to proceed otherwise. In particular, a logical question to ask is when has validation been successful enough to yield sufficient confidence in the M&S to apply the code to important stockpile problems? This is fundamentally a ***qualification*** question and it must be eventually answered.

Qualification is the process of establishing that the code supports the intended use for the DP customer. Qualification plays an important role for applications that are typically narrower than the stockpile driver for the V&V plan. The level of formality applied in qualification is well beyond the scope of this document. More importantly, precise definition of qualification requirements is a process that must be fundamentally owned by DP at Sandia. Nonetheless, the content requirements we place on Sandia code V&V plans in this document support code qualification.

The V&V peer review process defined in [Pilch] also explicitly aims to support code qualification activities, especially through the proposed Level 3 reviews discussed in that document. Figure 12 is taken from [Pilch] and illustrates a conception of how V&V and peer review can support new stockpile qualification efforts that rely to a greater or lesser extent on M&S capability from Sandia ASCI codes.

Given the need for qualification, the final content element for Sandia V&V plans that we discuss is guidance for using the code for important stockpile computing problems. This element addresses two concerns. The first concern is that the benefit of successfully executing a defined V&V plan and final qualification process can be completely negated through incorrect usage of the code. If only through self-defense on the part of the code team, there is great value in explicitly linking guidance for high-consequence application of the code to the V&V effort for the given stockpile driver.

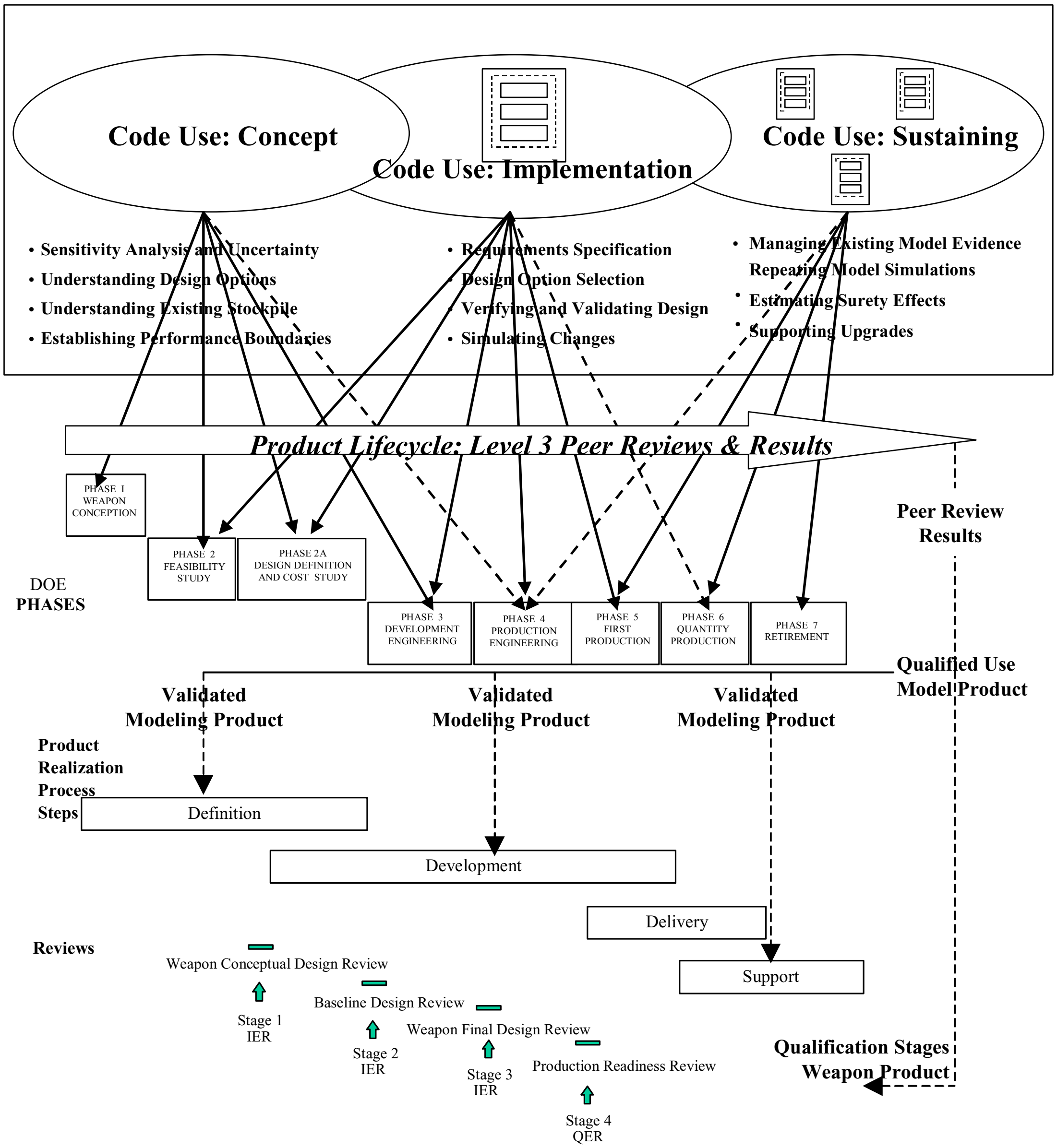

Figure 12. Role of code qualification in weapon qualification processes.

The second concern addresses who is most likely to provide correct and useful guidance for using the code. By placing usage guidance in proximity to the V&V planning content the code team will take advantage of the people who are most likely to be able to provide this information accurately for the chosen stockpile driver. Our conclusion is that the people who executed the V&V plan that supported the driver are in position to provide the best guidance for using the code for specific applications related to the stockpile driver.

8.2 Stockpile Computing Guidance

A long list of issues must be dealt with under the rubric of providing "stockpile computing guidance." From the perspective of a code user, guidance can be appropriate in many of the areas suggested in Figure 13. This figure is designed to be generic, but clearly is not generic enough to include all of the codes being developed under the ASCI program at Sandia. Figure 13 expresses two factors for performing stockpile computing that a DP customer might be concerned with. The first factor is that the appropriate level of ***technical expertise*** be applied when performing stockpile computing. The second factor is ***specific constraints*** associated with the formality of the stockpile problem that is being addressed by the code. The required code application could be exploratory and have few or no formal DP process constraints associated with it. Or, the required code application could be one part of a very formal stockpile process and involve very rigorous DP process requirements and constraints.

8.2.1 Guidance for Technical Expertise

Issues identified in Figure 13 that arise under the need for technical expertise in performing stockpile calculations include:

- Proper problem definition ("Input" in Figure 13).

- Proper execution of the "code," which could be more than one code. ("Code" in Figure 13).

- Proper analysis and accurate communication of results of the code application. ("Results" in Figure 13).

There is potentially a great deal of guidance that can be offered by the code team to address proper technical use of the code. Straightforward guidance includes such elements as:

- Reliance upon approved code documentation, such as User's Guides and theory manuals.

- Appropriate criteria for doing stockpile calculations should be designated.

- Minimum standards of user expertise for performing such calculations should be defined. This expertise might be established from formal training programs. Or, it may be decided that a user for a stockpile calculation must participate in confirmatory Tier IV validation activity.

- Specific scrutiny activities, such as input or output inspections, or both.

- Use of code execution and analysis environments (scripts, GUI's, or other interfaces).

- Use of independent multiple users or analysts to perform the required stockpile application of the code.

Whatever is recommended should be documented in this content element of the plan.

8.2.2 Guidance for Process Requirements

The other factor that influence stockpile use of the code is that there may be constraints that arise from requirements associated with the DP process that the calculation is part of. These constraints may take the form of specific archiving demands, specific procedures be applied, or other kinds of requirements that are associated with the specific stockpile task at hand. Little general guidance can be given to address this issue. Instead, it is clear that these constraints can only be identified and incorporated in the stockpile computing guidance through collaboration with the DP customer. However, if such constraints are known to be associated with the stockpile driver for the V&V plan ahead of time they should be identified in the plan.

Some examples of guidance that may likely fall in this area include:

- Formal specification of the qualified use of the code. This may include, but is not restricted to, discussion of construction and archiving of code inputs and outputs, guidance for mesh convergence studies, and material property sufficiency requirements for stockpile computing.

- Formal qualification of users for application of the code to stockpile problems.

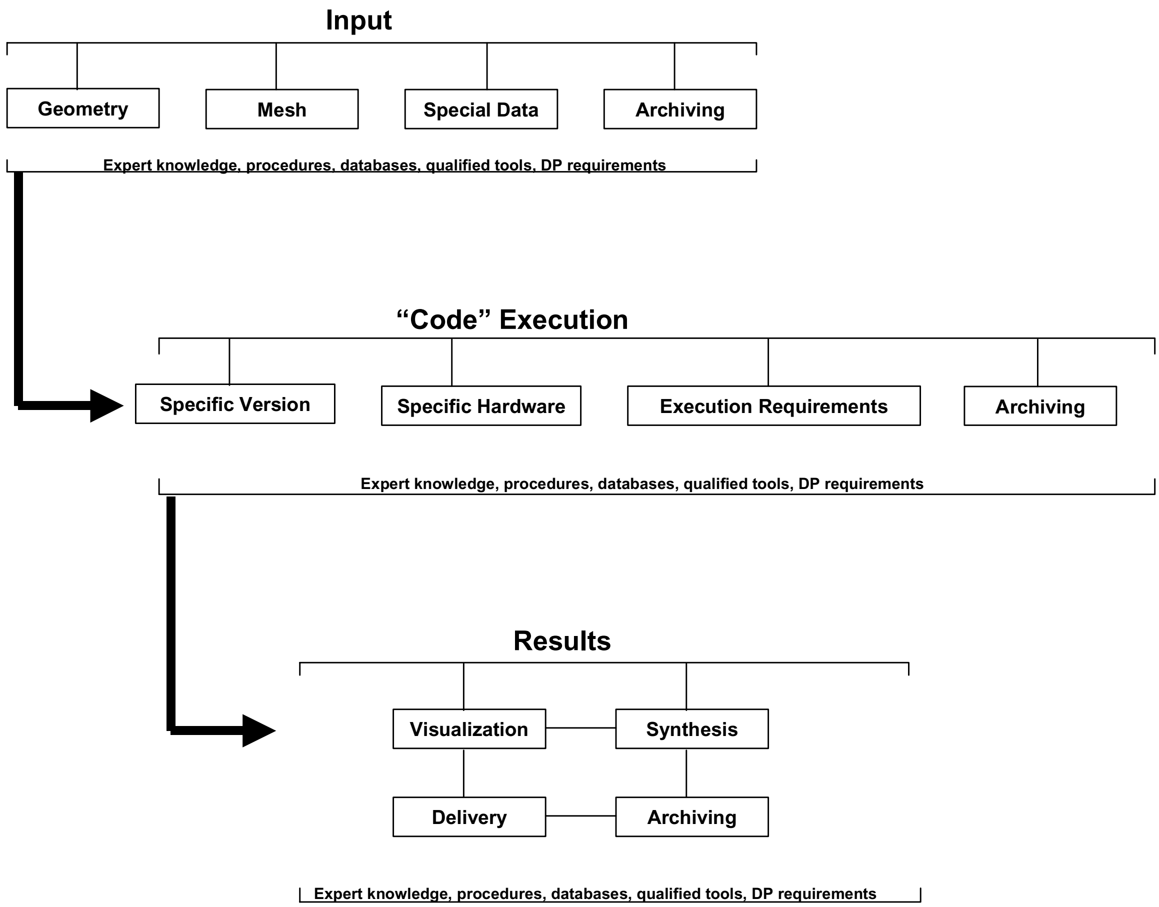

Figure 13. Needed work to perform stockpile computing with certain Sandia ASCI codes.

- Description of formal methods used for scrutinizing stockpile simulations with peer review.

- Description of methods used for auditing the simulation process to guarantee that test problems are successfully completed, configuration control is maintained, necessary information is archived, and documentation is completed.

8.3 Conclusion

An important conclusion we emphasize is that the needs discussed in this section do not constitute a novel approach to high-consequence computing. For example, the European Computational Fluid Dynamics (CFD) community has understood the need to move beyond V&V activities when performing high-consequence computation. [ERCOFTAC] is designed to be a quantum leap beyond the V&V guidance provided by the AIAA in [AIAA]. In [ERCOFTAC] a series of guidelines are developed for performing "industrial strength" CFD calculations. Figure 14 lists some of the guidelines discussed in [ERCOFTAC] that are of particular relevance to this section. The similarity is significant.

Guidelines on training
Guidelines on problem definition
Guidelines on geometry definition and gridding
Guidelines on equation solution
Guidelines on error assessment
Guidelines on interpretation
Guidelines on documentation
Guidelines on communication with code developers

Figure 14. Best practice advice for industrial CFD [ERCOFTAC].

8.4 Content Criteria Summary

The following summarizes the key content criteria reflecting the V&V plan elements discussed in this section.

GSC:	Guidance for stockpile computing using the code is provided.
GSC1:	Technical guidance for code application is provided.
GSC2:	Process guidance for code application to the associated stockpile driver is provided.

9. Conclusions

The purpose of this report is to state guidelines for content appropriate for the development and documentation of Verification and Validation plans for ASCI code development projects at Sandia. We have chosen to emphasize requirements driven V&V. This conscious choice has given the stockpile requirements which lie at the heart of the intended applications of the Sandia ASCI codes for SBSS the most important role in the entire planning process. Our view is most succinctly expressed in Figure 2, which shows the anticipated logic in the V&V plans flowing from the underlying stockpile driver to the required V&V activity.

With this approach, the content we recommend for Sandia V&V plans consists of four major items:

- Definition of the stockpile driver and associated requirements to be supported by modeling application of the code.

- Definition of the M&S requirements and physical phenomena required for modeling to successfully support the required stockpile driver and their relative importance to the modeling. A Phenomena Identification and Ranking Table (PIRT) is the chosen methodology for expressing this information.

- Definition of verification activities designed to assess and establish correctness of the software implementation of the phenomena in the PIRT.

- Definition of validation activities designed to assess and establish correctness of the implemented models that express the phenomena in the PIRT.

There are alternative strategies for developing V&V plans, such as specification-based V&V. We believe that our view, which stresses the ultimate importance of the stockpile applications for the code, is certainly more applicable for ASCI code development projects. We believe that our suggested approach is compatible with our view that we are ***validating codes for specific applications***, rather than attempting to validate codes in some entirety. We believe that the latter goal is impossible. On the other hand, focusing V&V activities to ultimately qualify the competence of a code for performing a specific set of modeling activities seems to us to be fully possible. Our recommended content guidelines in this report provide a basis for making such assessments.

(Page Left Blank)

References

[AIAA] American Institute of Aeronautics and Astonautics (1998), "Guide for the Verification and Validation of Computational Fluid Dynamics Simulations," AIAA-G-077-1998.

[Ambrosiano] J. Ambrosiano and M. – M. Peterson (2000), "Research Software Development Based on Exploratory Workflows: The Exploratory Process Model (ExP)," Los Alamos National Laboratory, LA-UR-00-3697.

[Boyack] B. E. Boyack, et al (1990), "Quantifying reactor safety margins Part 1: An overview of the code scaling, applicability, and uncertainty evaluation methodology," Nucl. Engng. Design, Vol. 119, 1-15.

[DMSO] Defense Modeling and Simulation Office (1996), "Verification, Validation, and Accreditation (VV&A) Recommended Practices Guide," Office of the Director of Defense Research and Engineering.

[DOE-2000a] U.S. Department of Energy Defense Programs (2000), "Accelerated Strategic Computing Initiative (ASCI) Program Plan," DOE/DP-99-000010592.

[DOE-2000b] U.S. Department of Energy Defense Programs (2000), "ASCI Program Software Quality Engineering: Goals, Principles, and Guidelines," DOE/DP/ASC--SQE-2000, to be published.

[ERCOFTAC] M. Casey and T. Wintergate, eds.(2000), "ERCOFTAC Special Interest Group on 'Quality and Trust in Industrial CFD' – Best Practices Guidelines," http://imhefwww.epfl.ch/ERCOFTAC.

[Hanson] G. Hanson, G. E. Wilson, M. G. Ortiz, and D. P. Griggs (1992), "Development of a Phenomena Identification and Ranking Table (PIRT) for a Postulated Double-Ended Guillotine Break in a Production Reactor," Nucl. Engng. Design, Vol. 136, 335-346.

[Hodges] A. Hodges, G. Froehlich, M. Pilch, and D. Peercy (2000), "Sandia National Laboratories ASCI Applications V&V Software Quality Engineering Guidelines: Practices and Work Products," Sandia National Laboratories, to be published.

[Johannes] J. Johannes (1999), personal communication.

[Kroeger] P. G. Kroeger, U. S. Rohatgi, J. H. Jo, and G. C. Slovik (1998), "Preliminary Phenomena Identification and Ranking Tables for simplified boiling water reactor loss-of-coolant accident scenarios," Brookhaven National Laboratory, NUREG/CR-6472.

[Larzelere] A. R. Larzelere (1998), "Simulation Development Capability," IEEE Computational Science & Engineering, January-March, 27-35.

[Lee] J. R. Lee (1998), "Certainty in Stockpile Computing: Recommending a Verification and Validation Program for Scientific Software", Sandia National Laboratories Report SAND98-2420.

[Pilch] M. Pilch, T. Trucano, D. Peercy, A. Hodges, E. Young, and J. Moya (2000), "Peer Review Process for the Sandia ASCI V&V Program: Version 1.0," Sandia National Laboratories, SAND2000-3099.

[PureCoverage] Pure Software, Inc. (1994), "PureCoverage: User's Guide," Sunnyvale, Cailfornia.

[Rohatgi] U. S. Rohatgi, H. S. Cheng, H. J. Khan, and W. Wulff (1997), "Preliminary Phenomena Identification and Ranking Tables (PIRT) for SBWR start-up stability," Brookhaven National Laboratory, NUREG/CR-6474.

[Saaty] T. Saaty (1982), Decision-Making For Leaders, Lifetime Learning Publications, Wadsworth, Inc., Belmont, California.

[Salari] K. Salari and P. Knupp (2000), "Code Verification by the Method of Manufactured Solutions," Sandia National Laboratories, SAND2000-1444.

[Shaw] R. A. Shaw, S. Z. Rouhani, T. K. Larson, and R. A. Dimenna (1988), "Development of a Phenomena Identification and Ranking Table (PIRT) for thermal-hydraulic phenomena during a PWR large-break LOCA," Idaho National Engineering Laboratory, NUREG/CR-5047.

[TPG] Technical Program Group (1989), "Quantifying reactor safety margins: Application of code scaling, applicability, and uncertainty evaluation methodology to a large-break, loss-of-coolant accident," Idaho National Engineering Laboratory, NUREG/CR-5249.

[Trucano] T. G. Trucano and J. L. Moya (1999), "Guidelines for Sandia ASCI Verification and Validation Plans – Content and Format: Version 1.0," Sandia National Laboratories, SAND99-3098.

[Wilson-1990] G. E. Wilson, et al (1990), "Quantifying reactor safety margins Part 2: Characterization of important contributors to uncertainty," Nucl. Engng. Design, Vol. 119, 17-31.

[Wilson-1997] G. E. Wilson, et al (1997), "Final Report: BWR drywell debris transport Phenomena Identification and Ranking Tables (PIRTs)," Idaho National Engineering and Environmental Laboratory, INEEL/EXT-97-00894.

[Wilson-1998] G. E. Wilson and B. E. Boyack (1998), "The role of the PIRT process in experiments, code development and code applications associated with reactor safety analysis," Nucl. Engng. Design, Vol. 186, 23-37.

[Woodward] P. Woodward and P. Colella (1984), "The numerical-simulation of two-dimensional fluid-flow with strong shocks," J. Comp. Phys., Vol. 54, No. 1, 115-173.

[Wulff] W. Wulff, et al (1990), "Quantifying reactor safety margins Part 3: Assessment and ranging of parameters," Nucl. Engng. Design, Vol. 119, 33-65.

[Zuber] N. Zuber, et al (1998), "An integrated structure and scaling methodology for severe accident technical issue resolution: Development of methodology," Nucl. Engng. Design, Vol. 186, 1-21.

(Page Left Blank)

Appendix A: Sandia ASCI V&V Plan Table of Contents

The recommended form of V&V plans for ASCI codes at Sandia is as follows:

Executive Summary (Optional)

Section 1: Introduction

PR1:	The V&V plan authorship includes the V&V process owner, and experimenter, and a DP customer representative.
PR2:	The plan is compatible with the format specified in the Version 2.0 guidelines.
PR3:	A single stockpile driver for the V&V plan is identified.
PR4:	The V&V planning process is described.

Section 2: Stockpile Drivers and DP Customer Requirements

PR3:	A single stockpile driver for the V&V plan is identified.
DP1:	The appropriate customers and constraints associated with the stockpile driver have been identified.
DP2:	The detailed stockpile requirements been extracted from the stockpile driver.
DP3:	The stockpile requirements have been mapped to M&S needs and requirements.
DP4:	There are sufficient requirements to allow the development of a useful PIRT.

Section 3: PIRT

PIRT1:	The PIRT is present in the V&V plan.
PIRT2:	The PIRT process methodology is described.
PIRT3:	Phenomena in the PIRT are ranked by a rational scoring system.
PIRT4:	Current M&S status (capability) for each of the phenomena in the PIRT is presented.

Section 4: SQE

SQE:	Content required by DOE and Sandia SQE policy documents is present.
SQE1:	Software V&V.
SQE2:	Software Engineering.
SQE3:	Project Management.

Section 5: Verification Test Suite

VERTS:	A Verification Test Suite has been constructed and documented.
VERTS1:	The structure and logic of the VERTS is addressed.
VERTS2:	The construction of the VERTS is addressed.
VERTS3:	The acceptance metrics for the VERTS are addressed.

Section 6: Validation Plan

VAL1:	The relation between validation activities and the PIRT is defined.
VAL2:	The data requirements associated with the validation activity are described.
VAL3:	The dependence of assessment of the fidelity of the code physical models upon experimental data is described.
VAL4:	The opportunities and obligation for experimental investigation of the assumptions underlying the implemented physical models are discussed.
VAL5:	The dependence of the code models upon experimentally measured quantities is discussed.
VAL6:	The prioritized experimental needs are described.
VAL7:	The anticipated use of technologies like uncertainty quantification in the validation activities is described.
VALTS:	A Validation Test Suite is constructed and documented.
VALTS1:	The structure of the VALTS is described. The structure is compatible with the PIRT and the four-tiered approach described in these guidelines.
VALTS2:	The construction of the VALTS is addressed.
VALTS3:	The acceptance metrics for the VALTS are addressed.

Section 7: Stockpile Computing Guidance

GSC:	Guidance for stockpile computing using the code is provided.
GSC1:	Technical guidance for code application is provided.
GSC2:	Process guidance for code application to the associated stockpile driver is provided.

References

Acknowledgements

The guidelines we publish in this document have been heavily influenced by the 1998 ASCI V&V Draft Program Plan, Version 1.0 of this document, and the 2000 prototype V&V peer review process at Sandia. We would like to thank Dan Carroll and Vicente Romero for critically reviewing this document prior to publication. We also thank Sheldon Tieszen, Walter Wolfe and Bill Oberkampf for critically reading the manuscript prior to publication. Many others too numerous to acknowledge individually have constructively commented on specific elements in this report.

This work was supported by the United States Department of Energy under Contract DE-AC04-94AL85000. Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy.

(Page Left Blank)

Distribution

EXTERNAL DISTRIBUTION

M. A. Adams
Jet Propulsion Laboratory
4800 Oak Grove Drive, MS 97
Pasadena, CA 91109

M. Aivazis
Center for Advanced Computing Research
California Institute of Technology
1200 E. California Blvd./MS 158-79
Pasadena, CA 91125

Charles E. Anderson
Southwest Research Institute
P. O. Drawer 28510
San Antonio, TX 78284

Mark Anderson
ACTA
2790 Skypark Dr., Suite 310
Torrance, CA 90505-5345

Bilal Ayyub (2)
Department of Civil Engineering
University of Maryland
College Park, MD 20742

Osman Balci
Department of Computer Science
Virginia Tech
Blacksburg, VA 24061

Steven Batill (2)
Dept. of Aerospace & Mechanical Engr.
University of Notre Dame
Notre Dame, IN 46556

S. Beissel
Alliant Techsystems, Inc.
600 Second St., NE
Hopkins, MN 55343

David Belk
WL/MNAA
101 W. Eglin Blvd., Suite 219
Eglin AFB, FL 32542-6810

Ted Belytschko (2)
Department of Mechanical Engineering
Northwestern University
2145 Sheridan Road
Evanston, IL 60208

Pavel A. Bouzinov
ADINA R&D, Inc.
71 Elton Avenue
Watertown, MA 02472

John A. Cafeo
General Motors R&D Center
Mail Code 480-106-256
30500 Mound Road
Box 9055
Warren, MI 48090-9055

K. Casteel (2)
University of Texas at El Paso
600 W. University Ave.
El Paso, TX 79968-0521

James C. Cavendish
General Motors R&D Center
Mail Code 480-106-359
30500 Mound Road
Box 9055
Warren, MI 48090-9055

Wei Chen
Dept. of Mechanical Engr. (M/C 251)
842 W. Taylor St.
University of Illinois at Chicago
Chicago, IL 60607-7022

Kyeongjae Cho (2)
Dept. of Mechanical Engineering
MC 4040
Stanford University
Stanford, CA 94305-4040

Thomas Chwastyk
U.S. Navel Research Lab.
Code 6304
4555 Overlook Ave., SW
Washington, DC 20375-5343

Hugh Coleman (2)
Department of Mechanical &
Aero. Engineering
University of Alabama/Huntsville
Huntsville, AL 35899

Raymond Cosner (2)
Boeing-Phantom Works
MC S106-7126
P. O. Box 516
St. Louis, MO 63166-0516

P. Cuniff
U.S. Army Soldier Systems Center
Kansas Street
Natick, MA 01750-5019

Frank Dean (2)
Strategic Systems Programs
Nebraska Avenue Complex
287 Somers Court NW, Suite 10041
Washington, DC 20393-5446

Department of Energy (4)
Attn: Kevin Greenaugh, DP-153
Paul Messina, DP-51
Juan Meza, DP-51
William Reed, DP-51
Department of Energy
1000 Independence Ave., SW
Washington, DC 20585

U. M. Diwekar (2)
Center for Energy and
Environmental Studies
Carnegie Mellon University
Pittsburgh, PA 15213-3890

David Dolling
Department of Aerospace Engineering
& Engineering Mechanics
University of Texas at Austin
Austin, TX 78712-1085

Isaac Elishakoff (2)
Dept. of Mechanical Engineering
Florida Atlantic University
777 Glades Road
Boca Raton, FL 33431-0991

Joseph E. Flaherty (2)
Dept. of Computer Science
Rensselaer Polytechnic Institute
Troy, NY 12181

John Fortna
ANSYS, Inc.
275 Technology Drive
Canonsburg, PA 15317

Roger Ghanem
Dept. of Civil Engineering
Johns Hopkins University
Baltimore, MD 21218

Mike Giltrud
Defense Threat Reduction Agency
DTRA/CPWS
6801 Telegraph Road
Alexandria, VA 22310-3398

James Glimm (2)
Dept. of Applied Math & Statistics
P138A
State University of New York
Stony Brook, NY 11794-3600

James Gran
SRI International
Poulter Laboratory AH253
333 Ravenswood Avenue
Menlo Park, CA 94025

Bernard Grossman (2)
Dept. of Aerospace &
Ocean Engineering
Mail Stop 0203
215 Randolph Hall
Blacksburg, VA 24061

Sami Habchi
CFD Research Corp.
Cummings Research Park
215 Wynn Drive
Huntsville, AL 35805

Raphael Haftka (2)
Dept. of Aerospace and Mechanical
Engineering and Engineering
Science
P. O. Box 116250
University of Florida
Gainesville, FL 32611-6250

Achintya Haldar (2)
Dept. of Civil Engineering
& Engineering Mechanics
University of Arizona
Tucson, AZ 85721

Tim Hasselman
ACTA
2790 Skypark Dr., Suite 310
Torrance, CA 90505-5345

George Hazelrigg
Division of Design, Manufacturing
& Innovation
Room 508N
4201 Wilson Blvd.
Arlington, VA 22230

Richard Hills
College of Engineering, MSC 3449
New Mexico State University
P. O. Box 30001
Las Cruces, NM 88003

F. Owen Hoffman (2)
SENES
102 Donner Drive
Oak Ridge, TN 37830

G. Ivy
Logicon R&D Associates
P.O. Box 92500
Los Angeles, CA 90009

Ralph Jones (2)
Sverdrup Tech. Inc./AEDC Group
1099 Avenue C
Arnold AFB, TN 37389-9013

Leo Kadanoff
Research Institutes Building
University of Chicago
5640 South Ellis Ave.
Chicago, IL 60637

J. Keremes
The Boeing Company
Rocketdyne Propulsion & Power
P. O. Box 7922
6633 Canoga Avenue
Canoga Park, CA 91309-7922

Hyoung-Man Kim
Boeing Company
M/S: ZC-01
502 Gemini Ave.
Houston, TX 77058

K. D. Kimsey
U.S. Army Research Laboratory
Weapons & Materials Research
Directorate
AMSRL-WM-TC 309 120A
Aberdeen Proving Gd, MD 21005-5066

B. A. Kovac
The Boeing Company
Rocketdyne Propulsion & Power
P. O. Box 7922
6633 Canoga Avenue
Canoga Park, CA 91309-7922

P. Krysl
Department of Computer Science
California Institute of Technology
1200 E. California Blvd./MS 256-80
Pasadena, CA 91125

W. K. Liu (2)
Northwestern University
Dept. of Mechanical Engineering
2145 Sheridan Road
Evanston, IL 60108-3111

Robert Lust
General Motors, R&D and Planning
MC 480-106-256
30500 Mound Road
Warren, MI 48090-9055

Sankaran Mahadevan (2)
Dept. of Civil &
Environmental Engineering
Vanderbilt University
Box 6077, Station B
Nashville, TN 37235

Hans Mair
Institute for Defense Analysis
Operational Evaluation Division
1801 North Beauregard Street
Alexandria, VA 22311-1772

W. McDonald
Naval Surface Warfare Center
Code 420
101 Strauss Avenue
Indian Head, MD 20640-5035

Gregory McRae (2)
Dept. of Chemical Engineering
Massachusetts Institute of Technology
Cambridge, MA 02139

Michael Mendenhall (2)
Nielsen Engineering & Research, Inc.
510 Clyde Ave.
Mountain View, CA 94043

Sue Minkoff (2)
Dept. of Mathematics and Statistics
University of Maryland, Baltimore Co.
1000 Hilltop Circle
Baltimore, MD 21250

Max Morris (2)
Department of Statistics
Iowa State University
304A Snedecor-Hall
Ames, IW 50011-1210

Paul Muessig
Naval Air Warfare Center
Joint Accreditation Support Activity
Weapons Division, Code 418000D
1 Administration Circle
China Lake, CA 93555-6100

R. Namburu
U.S. Army Research Laboratory
AMSRL-CI-H
Aberdeen Proving Gd, MD 21005-5067
NASA/Ames Research Center
Attn: U. B. Mehta
MS: T27 B-1
Moffett Field, CA 94035-1000

NASA/Glen Research Center
Attn: Chris Steffen, MS 5-11
Cleveland, OH 44135

NASA/Langley Research Center (5)
Attn: Michael Hemsch, MS 280
Jim Luckring, MS 280
Ahmed Noor, MS 369
Sharon Padula, MS 159
Jim Weilmuenster, MS 408A
Hampton, VA 23681-0001

C. Needham
Applied Research Associates, Inc.
4300 San Mateo Blvd., Suite A-220
Albuquerque, NM 87110

A. Needleman
Division of Engineering, Box D
Brown University
Providence, RI 02912

Robert Nelson (2)
Dept. of Aerospace &
Mechanical Engineering
University of Notre Dame
Notre Dame, IN 46556

Efstratios Nikolaidis (2)
4320 Pine Ridge Court
Ann Arbor, MI 48105

Tinsley Oden (2)
Texas Institute of Comp. Mechanics
University of Texas at Austin
Austin, TX 78712

Michael Ortiz (2)
Graduate Aeronautical Laboratories
California Institute of Technology
1200 E. California Blvd./MS 105-50
Pasadena, CA 91125

Dale Pace
Applied Physics Laboratory
Johns Hopkins University
111000 Johns Hopkins Road
Laurel, MD 20723-6099

Alex Pang
Computer Science Department
University of California
Santa Cruz, CA 95064

Allan Pifko
2 George Court
Melville, NY 11747

Cary Presser (2)
Process Measurements Div.
National Institute of Standards
and Technology
Bldg. 221, Room B312
Gaithersburg, MD 20899

P. Radovitzky
Graduate Aeronautical Laboratories
California Institute of Technology
1200 E. California Blvd./MS 105-50
Pasadena, CA 91125

W. Rafaniello
DOW Chemical Company
1776 Building
Midland, MI 48674

Chris Rahaim (2)
SAIC
14 East Washington St., Suite 401
Orlando, FL 32801-2320

Pradeep Raj (2)
Computational Fluid Dynamics
Lockheed Martin Aeronautical Sys.
86 South Cobb Drive
Marietta, GA 30063-0685

J. N. Reddy
Dept. of Mechanical Engineering
Texas A&M University
ENPH Building, Room 210
College Station, TX 77843-3123

John Renaud (2)
Dept. of Aerospace & Mechanical Engr.
University of Notre Dame
Notre Dame, IN 46556

E. Repetto
Graduate Aeronautical Laboratories
California Institute of Technology
1200 E. California Blvd./MS 105-50
Pasadena, CA 91125

Patrick J. Roache
1108 Mesa Loop NW
Los Lunas, NM 87031

A. J. Rosakis
Graduate Aeronautical Laboratories
California Institute of Technology
1200 E. California Blvd./MS 105-50
Pasadena, CA 91125

Tim Ross (2)
Dept. of Civil Engineering
University of New Mexico
Albuquerque, NM 87131

Sunil Saigal (2)
Carnegie Mellon University
Department of Civil and
Environmental Engineering
Pittsburgh, PA 15213

Len Schwer
Schwer Engineering & Consulting
6122 Aaron Court
Windsor, CA 95492

Paul Senseny
Factory Mutual Research Corporation
1151 Boston-Providence Turnpike
P.O. Box 9102
Norwood, MA 02062

E. Sevin
Logicon RDA, Inc.
1782 Kenton Circle
Lyndhurst, OH 44124

Mark Shephard (2)
Rensselaer Polytechnic Institute
Scientific Computation Research Center
Troy, NY 12180-3950

T. P. Shivananda
Bldg. SB2/Rm. 1011
TRW/Ballistic Missiles Division
P. O. Box 1310
San Bernardino, CA 92402-1310

Y.-C. Shu
Graduate Aeronautical Laboratories
California Institute of Technology
1200 E. California Blvd./MS 105-50
Pasadena, CA 91125

Don Simons
RDA Logicon
6053 West Century Blvd.
P.O. Box 92500
Los Angles, CA 90009

Munir Sindir
Boeing Company, MC IB39
Rocketdyne Propulsion & Power
P. O. Box 7922
6633 Canoga Avenue
Canoga Park, CA 91309-7922

Ashok Singhal (2)
CFD Research Corp.
Cummings Research Park
215 Wynn Drive
Huntsville, AL 35805

R. Singleton
Engineering Sciences Directorate
Army Research Office
4300 S. Miami Blvd.
P.O. Box 1221
Research Triangle Park, NC 27709-2211

W. E. Snowden
DARPA
7120 Laketree Drive
Fairfax Station, VA 22039

Bill Spencer (2)
Dept. of Civil Engineering and Geological Sciences
University of Notre Dame
Notre Dame, IN 46556-0767

D. E. Stevenson (2)
Computer Science Department
Clemson University
442 Edwards Hall, Box 341906
Clemson, SC 29631-1906

Tim Swafford
Sverdrup Tech. Inc./AEDC Group
1099 Avenue C
Arnold AFB, TN 37389-9013

Kenneth Tatum
Sverdrup Tech. Inc./AEDC Group
740 Fourth Ave.
Arnold AFB, TN 37389-6001

Ben Thacker
Southwest Research Institute
6220 Culebra Road
Postal Drawer 28510
San Antonio, TX 78228-0510

Ren-Jye Yang
Ford Research Laboratory
MD2115-SRL
P.O.Box 2053
Dearborn, MI 4812

Simone Youngblood
DOD/DMSO
Technical Director for VV&A
1901 N. Beauregard St., Suite 504
Alexandria, VA 22311

M. A. Zikry
North Carolina State University
Mechanical & Aerospace Engineering
2412 Broughton Hall, Box 7910
Raleigh, NC 27695

J. G. Zwissler
Jet Propulsion Laboratory
4800 Oak Grove Drive, MS 97-8
Pasadena, CA 91109-8099

FOREIGN DISTRIBUTION:

Yakov Ben-Haim (2)
Department of Mechanical Engineering
Technion-Israel Institute of Technology
Haifa 32000
ISRAEL

Graham de Vahl Davis (2)
CFD Research Laboratory
University of NSW
Sydney, NSW 2052
AUSTRIALIA

Luis Eca (2)
Instituto Superior Tecnico
Department of Mechanical Engineering
Av. Rovisco Pais
1096 Lisboa CODEX
PORTUGAL

Charles Hirsch (2)
Department of Fluid Mechanics
Vrije Universiteit Brussel
Pleinlaan, 2
B-1050 Brussels
BELGIUM

Igor Kozin
Systems Analysis Department
Riso National Laboratory
P. O. Box 49
DK-4000 Roskilde
DENMARK

K. Papoulia
Inst. Eng. Seismology & Earthquake
Engineering
P.O. Box 53, Finikas GR-55105
Thessaloniki
GREECE

Malcolm Wallace
National Engineering Laboratory
East Kilbride
Glasgow
G75 0QU
UNITED KINGDOM

Los Alamos National Laboratory (39)
Mail Station 5000
P.O. Box 1663
Los Alamos, NM 87545
Attn: Peter Adams, MS B220
Attn: Thomas Bement, MS F600
Attn: Terrence Bott, MS K557
Attn: D. Cagliostro, MS F645
Attn: David Crane, MS P946
Attn: John F. Davis, MS B295
Attn: Scott Doebling, MS P946
Attn: S. Eisenhawer, MS K557
Attn: Dawn Flicker, MS F664
Attn: George T. Gray, MS G755
Attn: Ken Hanson, MS P940
Attn: R. Henninger, MS D413
Attn: Brad Holian, MS B268
Attn: Kathleen Holian, MS B295
Attn: Darryl Holm, MS B284
Attn: James Hyman, MS B284
Attn: Michael E. Jones, MS B259
Attn: Cliff Joslyn, MS B265
Attn: James Kamm, MS D413
Attn: Jeanette Lagrange, MS D445
Attn: S. Keller-McNulty, MS F600
Attn: Elizabeth Kelly, MS F600
Attn: Ken Koch, MS F652
Attn: Len Margolin, MS D413
Attn: Harry Martz, MS F600
Attn: Mike McKay, MS F600
Attn: Mark P. Miller, MS P946
Attn: John D. Morrison, MS F602
Attn: Karen I. Pao, MS B256
Attn: M. Peterson-Schnell,
MS B295
Attn: William Rider, MS D413

Attn: Tom Seed, MS F663
Attn: David Sharp, MS B213
Attn: Richard N. Silver, MS D429
Attn: Ronald E. Smith, MS J576
Attn: Christine Treml, MS H851
Attn: Daniel Weeks, MS B295
Attn: Morgan White, MS F663
Attn: Alyson G. Wilson, MS F600

University of California (20)
Lawrence Livermore National Laboratory
7000 East Ave.
P.O. Box 808
Livermore, CA 94550
Attn: T. F. Adams, MS L-095
Attn: Steven Ashby, MS L-561
Attn: John Bolstad, MS L-023
Attn: Peter N. Brown, MS L-561
Attn: T. Scott Carman, MS L-031
Attn: R. Christensen, MS L-160
Attn: Richard Klein, MS L-023
Attn: Byung S. Lee, MS L-550
Attn: Kirk Levedahl, MS L-016
Attn: Roger Logan, MS L-125
Attn: C. F. McMillan, MS L-098
Attn: C. Mailhiot, MS L-055
Attn: J. F. McEnerney, MS L-023
Attn: G. M. Murphy, MS L-650
Attn: Daniel Nikkel, MS L-342
Attn: Cynthia Nitta, MS L-096
Attn: Douglas Post, MS L-038
Attn: Peter Raboin, MS L-125
Attn: Peter Terrill, MS L-125
Attn: Charles Tong, MS L-560

Argonne National Laboratory
Attn: Paul Hovland
MCS Division
Bldg. 221, Rm. C-236
9700 S. Cass Ave.
Argonne, IL 60439

SANDIA INTERNAL

1	MS 1152	1642	M. L. Kiefer
1	MS 1186	1674	R. J. Lawrence
1	MS 0525	1734	P. V. Plunkett
1	MS 0525	1734	R. B. Heath
1	MS 0525	1734	S. D. Wix
1	MS 1393	1902	J. R. Garcia
1	MS 0457	2001	W. J. Tedeschi
1	MS 0429	2100	J. S. Rottler
1	MS 0453	2104	D. L. McCoy
1	MS 0475	2105	R. C. Hartwig
1	MS 1393	2106	F. F. Dean
1	MS 0447	2111	J. O. Harrison
1	MS 0447	2111	P. D. Hoover
1	MS 0479	2151	P. A. Sena
1	MS 0479	2151	M. H. Abt
1	MS 0482	2161	V. J. Johnson
1	MS 0482	2161	R. S. Baty
1	MS 0481	2167	M. A. Rosenthal
1	MS 0481	2167	W. C. Moffatt
1	MS 0481	2168	K. D. Meeks
1	MS 0481	2168	K. Ortiz
1	MS 0509	2300	M. W. Callahan
1	MS 0769	5800	D. S. Miyoshi
1	MS 0759	5845	I. V. Waddoups
1	MS 0759	5845	M. S. Tierney
1	MS 0782	5861	R. V. Matalucci
1	MS 1201	5903	J. M. McGlaun
1	MS 0737	6114	P. A. Davis
1	MS 0751	6117	L. S. Costin
1	MS 0718	6141	C. D. Massey
1	MS 0708	6214	P. S. Veers
1	MS 0747	6410	R. L. Camp
1	MS 0747	6410	G. D. Wyss
1	MS 0746	6411	R. M. Cranwell
1	MS 0746	6411	D. J. Anderson
1	MS 0746	6411	J. E. Campbell
1	MS 0746	6411	D. G. Robinson
1	MS 0746	6411	L. P. Swiler
1	MS 1140	6500	J. K. Rice
1	MS 0974	6522	G. D. Valcez
1	MS 0977	6524	W. R. Cook
3	MS 0977	6524	S. M. DeLand
1	MS 1138	6533	E. Shepherd

1	MS 1138	6534	L. M. Claussen
25	MS 1137	6534	A. L. Hodges
1	MS 1138	6534	M. T. McCornack
1	MS 1137	6534	S. V. Romero
25	MS 1137	6535	G. K. Froehlich
1	MS 0771	6805	P. G. Kaplan
1	MS 1395	6821	J. W. Garner
1	MS 1395	6821	P. Vaughn
1	MS 0779	6849	J. C. Helton
1	MS 0779	6849	R. P. Rechard
1	MS 0779	6849	M. J.Shortencarier
1	MS 0778	6851	G. E. Barr
1	MS 0778	6851	R. J. MacKinnon
1	MS 9051	8351	C. A. Kennedy
1	MS 9202	8418	W. P. Ballard
1	MS 9202	8418	R. M. Zurn
1	MS 9405	8700	T. M. Dyer
1	MS 9042	8725	W. A. Kawahara
1	MS 9405	8726	R. E. Jones
1	MS 9161	8726	P. A. Klein
1	MS 9161	8726	E. P. Chen
1	MS 9042	8728	C. D. Moen
1	MS 9405	8743	R. A. Regueiro
1	MS 9003	8900	K. E. Washington
1	MS 9012	8920	P. E. Nielan
1	MS 9217	8950	P. T. Boggs
1	MS 9217	8950	P. D. Hough
1	MS 9217	8950	M. L. Koszykowski
1	MS 1110	8950	L. J. Lehoucq
1	MS 9217	8950	K. R. Long
1	MS 0841	9100	T. C. Bickel
1	MS 0828	9102	R. K. Thomas
1	MS 0841	9102	J. A. Fernandez
1	MS 0835	9111	S. N. Kempka
1	MS 0835	9111	S. P. Burns
1	MS 0835	9111	R. J. Cochran
1	MS 0835	9111	D. K. Gartling
1	MS 0835	9111	B. Hassan
1	MS 0835	9111	W. P. Wolfe
1	MS 0834	9112	A. C. Ratzel
1	MS 0826	9113	W. Hermina
1	MS 0826	9113	T. J. Bartel
1	MS 0827	9114	J. E. Johannes
1	MS 0827	9114	K. S. Chen
1	MS 0827	9114	L. A. Mondy
1	MS 0827	9114	R. R. Rao
1	MS 0827	9114	P. R. Schunk
3	MS 0825	9115	W. H. Rutledge
3	MS 0825	9115	S. J. Beresh
3	MS 0825	9115	F. G. Blottner
3	MS 0825	9115	D. W. Kuntz
1	MS 0825	9115	M. A. McWherter-Payne
3	MS 0825	9115	J. L. Payne
3	MS 0825	9115	D. L. Potter
1	MS 0825	9115	K. Salari
1	MS 0825	9115	L. W. Young
1	MS 0836	9116	E. S. Hertel
1	MS 0836	9116	M. R. Baer
1	MS 0836	9116	L. A. Gritzo
1	MS 0836	9116	S. R. Tieszen
1	MS 0827	9117	R. O. Griffith
1	MS 0367	9117	R. J. Buss
1	MS 0827	9117	R. B. Campbell
1	MS 0827	9117	D. Dobranich
1	MS 0836	9117	R. E. Hogan
1	MS 0827	9117	T. E. Voth
1	MS 0835	9121	J. S. Peery
1	MS 0835	9121	S. W. Attaway
1	MS 0835	9121	M. L. Blanford
1	MS 0835	9121	M. W. Heinstein
1	MS 0835	9121	S. W. Key
1	MS 0835	9121	G. M. Reese
1	MS 0555	9122	M. S. Garrett
1	MS 0847	9123	H. S. Morgan
1	MS 0847	9123	J. B. Aidun
1	MS 0847	9123	A. F. Fossum
1	MS 0847	9124	D. R. Martinez
3	MS 0847	9124	K. F. Alvin
1	MS 0847	9124	T. B. Carne
1	MS 0847	9124	J. L. Dohner
1	MS 0847	9124	R. V. Field
1	MS 0847	9124	D. O. Smallwood
1	MS 0557	9125	T. J. Baca
1	MS 0553	9126	R. A. May
1	MS 0827	9131	J. D. Zepper
1	MS 0827	9131	K. M. Aragon
1	MS 0827	9131	G. D. Sjaardema

1	MS 0827	9131	J. R. Stewart
25	MS 0828	9132	J. L. Moya
1	MS 0828	9132	S. N. Burchett
1	MS 0828	9132	K. V. Chavez
1	MS 0828	9132	T. Y. Chu
1	MS 0828	9132	C. W. Peterson
25	MS 0828	9133	M. Pilch
1	MS 0828	9133	B. F. Blackwell
1	MS 0828	9133	K. J. Dowding
1	MS 0828	9133	A. R. Lopez
1	MS 0828	9133	K. E. Metzinger
3	MS 0828	9133	W. L. Oberkampf
1	MS 0828	9133	T. L. Paez
1	MS 0828	9133	C. Romero
1	MS 0828	9133	V. J. Romero
1	MS 0828	9133	A. Urbina
1	MS 0828	9133	W. R. Witkowski
1	MS 1135	9134	D. B. Davis
1	MS 1135	9134	J. T. Nakos
1	MS 0321	9200	W. J. Camp
1	MS 1110	9211	D. E. Womble
1	MS 1110	9211	R. Carr
1	MS 1110	9211	S. Y. Chakerian
1	MS 0847	9211	M. S. Eldred
1	MS 0847	9211	A. A. Giunta
1	MS 1110	9211	W. E. Hart
1	MS 1110	9211	A. Johnson
1	MS 1110	9211	V. J. Leung
1	MS 1110	9211	C. A. Phillips
1	MS 0847	9211	J. R. Red-Horse
1	MS 0847	9211	B. Schimel
25	MS 0819	9211	T. G. Trucano
1	MS 0847	9211	B. A. vanBloemen Waanders
1	MS 0318	9212	G. S. Davidson
1	MS 1109	9212	R. J. Pryor
1	MS 1110	9214	J. DeLaurentis
1	MS 1110	9214	R. B. Lehoucq
1	MS 0321	9220	A. L. Hale
1	MS 1110	9223	N. D. Pundit
1	MS 0321	9224	J. A. Ang
1	MS 0321	9224	R. E. Benner
1	MS 0321	9224	J. L. Tompkins
1	MS 0847	9226	R. W. Leland
1	MS 0847	9226	B. A. Hendrickson
1	MS 0847	9226	P. Knupp
1	MS 0318	9227	P. D. Heermann
1	MS 0318	9227	C. F. Diegert
1	MS 0310	9232	P. Yarrington
1	MS 0819	9231	E. A. Boucheron
1	MS 0819	9231	K H. Brown
1	MS 0819	9231	K. G. Budge
1	MS 0819	9231	D. E. Carroll
1	MS 0819	9231	R. R. Drake
1	MS 0819	9231	A. C. Robinson
1	MS 0819	9231	R. M. Summers
1	MS 0819	9231	J. R. Weatherby
1	MS 0819	9231	M. K. Wong
1	MS 0820	9232	P. F. Chavez
1	MS 0820	9232	R. M. Brannon
1	MS 0820	9232	M. E. Kipp
1	MS 0820	9232	S. A. Silling
1	MS 0820	9232	P. A. Taylor
1	MS 0316	9233	S. S. Dosanjh
1	MS 1111	9221	S. J. Plimpton
1	MS 1111	9221	A. G. Salinger
1	MS 1111	9221	J. N. Shadid
1	MS 0316	9235	G. S. Heffelfinger
1	MS 1111	9225	H. P. Hjalmarson
1	MS 0660	9519	D. S. Eaton
1	MS 0660	9519	M. A. Ellis
1	MS 0419	9800	R. G. Easterling
1	MS 0421	9814	J. M. Sjulin
1	MS 0423	9817	R. A. Paulsen
1	MS 0423	9817	S. E. Dingman
1	MS 9003	9900	D. L. Crawford
1	MS 9003	9904	J. E. Kelly
1	MS 0428	12300	D. D. Carlson
1	MS 0490	12331	J. A. Cooper
1	MS 0829	12323	F. W. Spencer
1	MS 0829	12323	M. L. Abate
1	MS 0829	12323	B. M. Rutherford
1	MS 0638	12326	M. A. Blackledge
25	MS 0638	12326	D. E. Peercy
1	MS 0638	12326	D. L. Knirk
1	MS 0490	12331	P. E. D’Antonio
1	MS 0492	12332	D. R. Olson

1	MS 0405	12333	T. R. Jones
1	MS 0434	12334	R. J. Breeding
3	MS 0829	12335	K. V. Diegert
1	MS 1221	15002	R. D. Skocypec
1	MS 1179	15340	J. R. Lee
1	MS 1179	15341	L. J. Lorence
1	MS 0301	15400	J. L. McDowell
1	MS 9018	8945-1	Central Technical Files
2	MS 0899	9616	Technical Library
1	MS 0612	9612	Review & Approval Desk for DOE/OSTI