SANDIA REPORT

SAND2011-6677
Unlimited Release
Printed September 2011

A Design for a V&V and UQ *Discovery* Process

Patrick M. Knupp and Angel Urbina

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

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly-owned subsidiary of Lockheed Martin Corporation, for the U. S. Department of Energy's National Nuclear Security Administration under Contract DE-AC04-94AL85000.

Approved for public release; further dissemination unlimited.

Sandia National Laboratories

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

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online ordering: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2011-6677
Unlimited Release
Printed September 2011

A Design for a V&V and UQ *Discovery* Process

Patrick Knupp* and Angel Urbina†

Abstract

There is currently sparse literature on how to implement systematic and comprehensive processes for modern V&V/UQ (VU) within large computational simulation projects. Important design requirements have been identified in order to construct a viable "system" of processes. Significant processes that are needed include discovery, accumulation, and assessment. A preliminary design is presented for a VU Discovery process that accounts for an important subset of the requirements. The design uses a hierarchical approach to set context and a series of place-holders that identify the evidence and artifacts that need to be created in order to tell the VU story and to perform assessments. The hierarchy incorporates VU elements from a Predictive Capability Maturity Model and uses questionnaires to define critical issues in VU. The place-holders organize VU data within a central repository that serves as the official VU record of the project. A review process ensures that those who will contribute to the record have agreed to provide the evidence identified by the Discovery process. VU expertise is an essential part of this process and ensures that the roadmap provided by the Discovery process is adequate.

Both the requirements and the design were developed to support the Nuclear Energy Advanced Modeling and Simulation Waste project, which is developing a set of advanced codes for simulating the performance of nuclear waste storage sites. The Waste project served as an example to keep the design of the VU Discovery process grounded in practicalities. However, the system is represented abstractly so that it can be applied to other M&S projects.

*Uncertainty Quantification Dept. 1441, MS 1318, pknupp@sandia.gov
†Validation and Uncertainty Quantification Processes, Dept. 1544, MS 0828, aurbina@sandia.gov

Acknowledgment

This document was written with the support of the NEAMS VU management team, specifically Jim Stewart and Laura Swiler. We worked closely with the members of the Waste IPSC infrastructure team including Julie Bouchard, Carter Edwards, Vincent Mousseau, and Dena Vigil. We also worked with the Waste team modelers including Lupe Arguello, Geoff Freeze, Carlos Jove-Colon, Peter Schultz, and Yifeng Wang. Conversations with William Rider, William Stubblefied, and Tim Trucano were also valuable.

Contents

List of Figures

List of Tables

1 Introduction

This work was supported by the Nuclear Energy Advanced Modeling and Simulation (NEAMS) project, specifically by the NEAMS VU crosscut effort. The work, while focused on the abstract, was informed by the concrete example provided by close collaboration with members of the NEAMS Waste IPSC.

Computational simulation has flourished over the past two decades, driven in part by remarkable improvements in both hardware, algorithms, and software. Over the same period there has been a great deal of research on enabling the results of modeling & simulation to be predictive, i.e., to be able to quantitatively predict simulation outcomes before the fact, rather than after-the-fact. Unfortunately, technical advances in modern verification, validation, and uncertainty quantification (VU) methodology have tended to be under-applied in practice, particularly in large modeling and simulation projects. In fact, Oberkampf and Roy have said (see [1], Chapter 5)

"... our experience and the experience of others has convinced us that while technical issues and computing resources are important, they are not the limiting factor in improving the credibility and usefulness of scientific computing used in a decision-making environment."

They go on to say that non-technical issues have significantly constrained improvements in the amount and quality of information produced by simulations.

A major non-technical constraint has been the fact that modeling and simulation has existed well before the advent of modern VU methodology. This includes early VU methodology that is well established within some large simulation projects. A prime example is the Yucca Mountain project, which used Quality Assurance-based VU practices in order to meet the requirements of licensing agencies. The VU practices employed on many large simulation projects have not caught up with the explosive growth of knowledge in the field of VU and there is thus room for improvement. Modern VU practices are described in [1], [2], [3], and many other references. Although there have been some demonstrations of what modern VU has to offer within the context of a large simulation projects, it has had little or no systematic application. Without successful examples of systematic application of modern VU, it is difficult for others to deviate from their traditional tried-and-true modeling and simulation activities.

The present work is aimed at alleviating some barriers to the systematic application of modern VU within large simulation projects. One barrier exists because there has been no complete description of how one would do this. Systematic application requires a re-organization in the way in which VU information is gathered, stored, and disseminated. Partly, improvements in VU have been constrained because there exists a large data management component to implementing VU in a large project. This was recognized by the NEAMS Waste Integrated Performance and Safety Code development project (Waste IPSC). The V&V Plan written for this project specifically called for the development of an evidence management system which could be used to track modeling & simulation data and artifacts including those produced by VU activities [4]. Existence of such a management system (or an adequate substitute) would greatly facilitate the methods proposed in this report.

This report outlines the first steps in a method for systematically integrating modern VU methodology into a large simulation project. Specifically, we focus on what we call the *'Discovery'* process in which one creates place-holders for VU data that is arranged in a hierarchical fashion to provide context. Placeholders identify VU issues and activities specific to the modeling and simulation capability development sequence. As the work progresses, the placeholders are replaced with concrete data concerning the VU aspects of the project, thus enabling processes such as interim assessments to operate on the data.

Collaboration with the Waste project confronted us with many issues in attempting to 'discover' what were the VU aspects of the project and how to organize them. Examples of how our Discovery process applies to the Waste project are given in this report. We expect that our abstraction of the Discovery process will enable its use on other large M&S projects including the NEAMS Fuels and Reactor IPSC's. We also expect that the process will slowly be improved as a result of lessons-learned from applying it to other projects. In that sense, this is a living document.

The Discovery process is not the only process of importance in managing the VU aspects of a modeling and simulation project. Additional processes are also critical. For example, we will need an accumulation process for updating the data as new information is obtained (or plans are changed). We will also need an evaluation process for performing interim assessments that provide useful feedback to the developers. Ultimately, our goal is to develop a collection of processes that form a consistent *system* of processes that provide timely, well-organized information and documentation concerning verification, validation, and uncertainty quantification related to a particular modeling and simulation effort to the various stakeholders including external customers, project managers, modelers, and software developers. A description of processes in addition to Discovery is reserved for a follow-on report that will be based on the hierarchy/placeholder approach presented here.

With an eye toward a future description of a VU *assessment* process, our discovery process includes elements from the Predictive Capability Maturity Model (PCMM). The PCMM entails concepts for evaluating predictive capability within computational modeling and simulation, with the intent of improving information transparency, relevance, comprehensiveness, and communicability [5]. Evaluation and assessment is not intended to translate into a single quantitative measure, but rather into a nuanced view of the evidence produced to support claims of predictability. PCMM spans six content areas covering Modeling, Verification, Validation, and Uncertainty Quantification, each of which is further subdivided to provide a comprehensive view of the kinds of issues which should be considered in order to perform 'due diligence' and to build confidence in specific application-centric modeling and simulation work. The latest thinking about the PCMM can be found in [6].

As the title of this report indicates, we are proposing a design for the Discovery phase in VU. A design is not automatically a solution. The authors acknowledge that they do not have the foresight and wisdom to propose a design that is guaranteed to account for everything that is needed right out-of-the-box. What is needed beyond a design is an implementation within a large M&S project in order to test and improve the design. The Waste project served in that capacity for this initial design and we thank them for giving us this opportunity.

2 Requirements for a System of VU Processes

Any set of VU processes must address a list of requirements that describe in general terms what the processes must do. Many of these requirements can be found in the literature [1]. In the course of working with the Waste team, we identified a number of additional requirements, resulting in the following *working* list of requirements that the system of processes should collectively satisfy. The list contains a large number of requirements (30 to be exact), which may partly account for the fact that there have been few (if any) attempts at defining processes that address these requirements in total. The list is probably not definitive, but can serve as a starting point for developing a system of processes.

As will be explained in Section 9, the Discovery process presented in this report addresses, at least in part, the first fifteen of the requirements below (I(a) through I(o)) and lays the groundwork for addressing many of the other requirements. Requirements II(a) to II(g) are tentatively assigned to be addressed by an Accumulation Process to be described in a follow-on report, in which one accumulates evidence and helps manage the on-going VU work. Requirements III(a) to III(h) are tentatively assigned to be addressed by a graded, interim Assessment process which provides a status summary and feedback to the modelers and developers and, ultimately, to external stake-holders.

Requirements for a System of VU processes:

I. Requirements related to the Discovery process:

   (a) *Planning.* The system of processes must assist in the planning of VU work.

   (b) *Sequencing.* The system must indicate the order in which the work proceeds and flag any work that is done out of order.

   (c) *Terminology.* The system must use clear terminology to avoid confusion and provide unambiguous information.

   (d) *Context.* The system must provide context to define and manage the relationships between components and data within the system.

   (e) *Initiation.* The system must be able to be installed at any point in time within an on-going project and account for any VU work already performed.

   (f) *Consonance.* The system must promote appropriate practices within VU, such as those defined in [3] and other VU references.

   (g) *Concurrent.* The system must be concurrent with other activities that occur as part of the development and use of the modeling and simulation capability.

   (h) *Practical.* The system must be practical, as demonstrated by its use on at least one concrete example.

(i) *Transferable.* The system must be sufficiently general so that it can be applied with ease to more than one concrete example.

(j) *Deliberate.* The system must ensure that the VU work is done deliberately and thoughtfully, as opposed to haphazardly.

(k) *Multiple Use-Cases.* The system must account for different use-cases that occur within VU.

(l) *Transparency.* The system must be transparent in that the process and its interim results are public.

(m) *Records.* The system must identify the appropriate data, evidence and documentation that is to be saved and managed over time.

(n) *VU Requirements.* The system must call out the VU requirements that are to be met.

(o) *Traceability.* The system must provide traceability between VU requirements and the evidence.

II. Requirements related to the Accumulation process

(a) *Accumulation.* The system must ensure that the right data and evidence is accumulated.

(b) *Well-defined.* The system must be well-defined so that one always knows where in the processes one is and what the next step is. Most situations that can occur are accounted for.

(c) *Human Error.* The system must ensure that sources of human error are controlled.

(d) *Reusable.* The system must allow for reuse of relevant work and results when new models, new applications, or new plans are introduced.

(e) *Flexible.* The system must be able to update the data within the evidence management system pertaining to the evidence and documentation being accumulated.

(f) *Justified Choices.* The system must require justifications for choices made.

(g) *Interpret-ability.* The system must be able to synthesize interim and final raw evidence into a form that can be readily assimilated by team members, managers, and external stakeholders.

III. Requirements related to the Assessment process

(a) *Status.* The system must be able to provide interim status reports concerning the VU tasks within the project.

(b) *Assessments.* The system must provide interim, nuanced (graded) assessments of the work.

(c) *Accuracy.* The system must ensure that model accuracy and quality are appropriate.

(d) *Gaps & Weaknesses*. The system must identify gaps and weaknesses, as well as be able to suggest remediation's thereof.

(e) *Balance*. The system must provide a mechanism for balancing the work, so that gaps and weak points are addressed rather than over-emphasizing work on strong points.

(f) *Economical*. The system must ensure that no more work is done than necessary to meet the requirements.

(g) *Usable*. The system must make clear how the final results are to be used.

(h) *Completion*. The system must provide a method for deciding when it is appropriate to terminate.

2.1 Our Specific Discovery Process

Although the requirements under Discovery above give some idea of what a discovery process should do, we need to characterize the specific Discovery process presented in this report in terms of what is it that we want to 'discover' and why. The answer is, we want to identify a useful and specific VU-oriented contextual framework that accurately reflects the logical structure of VU within the project in its current and future states. The framework consists of hierarchically arranged entities of VU significance. Within each of the entities in the hierarchy are place-holders for 'VU data' that is placed under configuration control within a computer repository. We therefore also want to 'discover' (identify) the specific place-holders for VU data within the project. The Discovery process establishes a hierarchy that provides a contextual road-map for planning VU activities, accumulating & maintaining VU data, and which permits nuanced and timely assessments of the VU aspects of the project. The framework organizes the data within the central project repository and so becomes the official VU record of the project. One of the most important things that the Discovery process does, in the course of establishing the specific framework, is help establish a common terminology which everyone on the project can use. The terminology is based on modern VU concepts and that is one of the distinguishing features of our approach. The next chapter establishes some of the terminology and concepts that we will need.

3 Simulation Hierarchies

In order to initiate the description of the Discovery process, this chapter provides a description and terminology for certain modeling and simulation hierarchies that provide context for VU. We describe hierarchies for simulation capabilities, simulations, and simulation ensembles. This terminology sets the stage for the introduction of other modeling and simulation concepts and hierarchies in the next chapter.

3.1 Simulation Capabilities

Two important activities within modeling and simulation are the development of mathematical models and their associated data, and the implementation of those models within software. The goal of these activities (and others) is to develop what is termed a *simulation capability*. Figure 1 shows the conceptual parts that constitute a simulation capability. A simulation capability consists of two parts: *simulation software* and a set of *math models* describing systems, effects, and behavior. Conceptually, the software contains an implementation of the math models. The math models consist of a set of *local models* which (along with their data) describe in detail smaller portions of a *global math model*. In turn, the global math model describes, in terms of integration, the complete conceptual model and system of equations which the software is intended to simulate. The system may be a physical system but could potentially correspond to some other system which is to be simulated. The simulation software may contain: algorithms for solving the global math model, implementations of the local math models, algorithms for solving the local math models, and infrastructure or other supporting code. The math models describing the local or global system consists of two parts: the mathematical relations and the associated data and/or parameter values that complete the model.

Because simulation software may contain multiple local math models, we need to distinguish between *relevant math models* (and internal capabilities) from those which are irrelevant. This distinction is most commonly accomplished by a formal description of the *Intended Use* of the capability. Models which contribute toward the intended use are relevant models. In the remainder of this document, we will only be concerned with the relevant math models, and VU methods within the project are applied only to the relevant models. Therefore, the local models shown in Figure 1 are only the relevant models. Relevant models can be old models inherited from past projects or they can be new models which are developed under the current project. Inherited models can also undergo development within the current project.

Some of the parts within a simulation capability are optional. For example, a subcontinuum simulation capability may not contain any local math models. For a PA code, the simulation capability might consist of both the simulation software, the math models, and a UQ driver code. However, every simulation capability must contain simulation software and a global math model. Although this conceptual description of a simulation capability may differ somewhat from an actual example, it will serve as the starting point for describing additional hierarchies of importance.

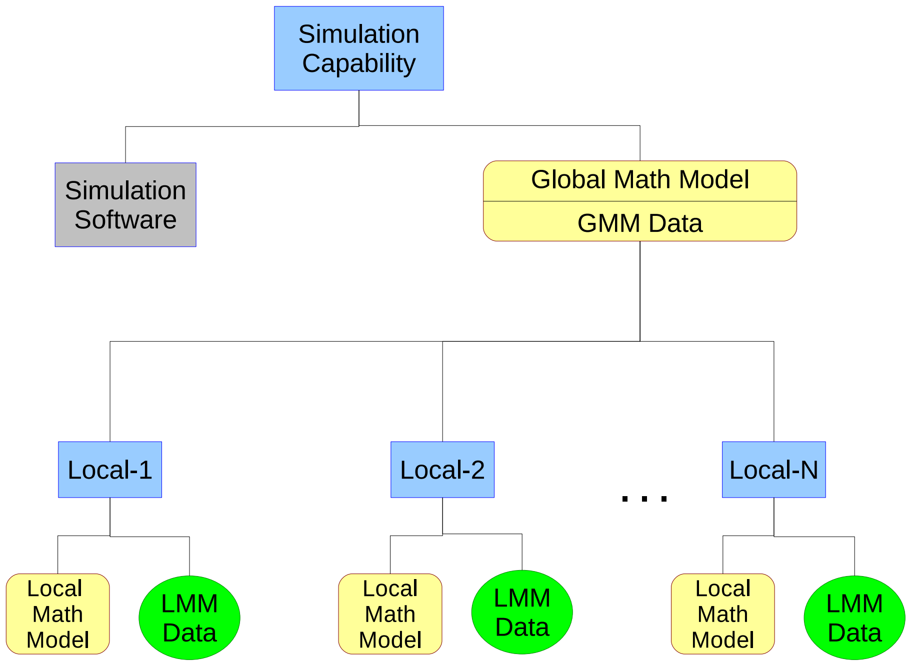

Figure 1. Conceptual Parts of a Simulation Capability

The following information may be required to define a specific simulation capability:

1. The name of the simulation capability, a version number, and a date corresponding to when the particular version of the capability was created.

2. The names and version numbers associated with the simulation software.

3. The name and version number of each math model, and the equations themselves.

Example of a Simulation Capability from Waste. Figure 2 depicts the Glass-Brine Dissolution Simulation Capability from Waste. The capability allows one to calculate concentrations of various components within the glass-brine solution at chemical equilibrium. The simulation software is called the Cantera Interface (or driver) code, which defines the specific components within the solution and calls the Cantera

Library. Specifically, the VCS solver within the Library is called. The solver minimizes the Gibbs' Free Energy equations, which constitute the Global Math Model. In turn, coefficients in the Global Math Model are obtained from the Yucca Mountain Thermodynamic Database. There are no local math models in this simulation capability.

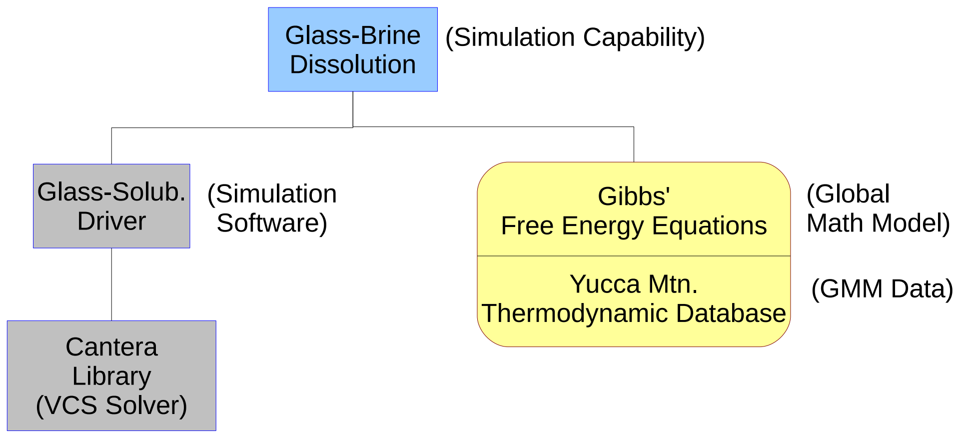

Figure 2. The Waste Glass-Brine Chemical Equilibrium Simulation Capability

3.2 Landmark Simulation Capabilities

A 'landmark' simulation capability is a simulation capability to which a VU objective has been assigned (the concept of a VU objective is discussed in Section 6.1). For now, suffice it to say that a landmark simulation capability is a particular version of a simulation capability which has significance from the VU-perspective. The hierarchy for a landmark simulation capability is the same as for a simulation capability, except that an additional box is added directly below the simulation capability box, on the same level as the simulation software and global math model boxes. Figure 3 shows the conceptual parts of a landmark simulation capability. A landmark simulation capability may have more than one VU objective.

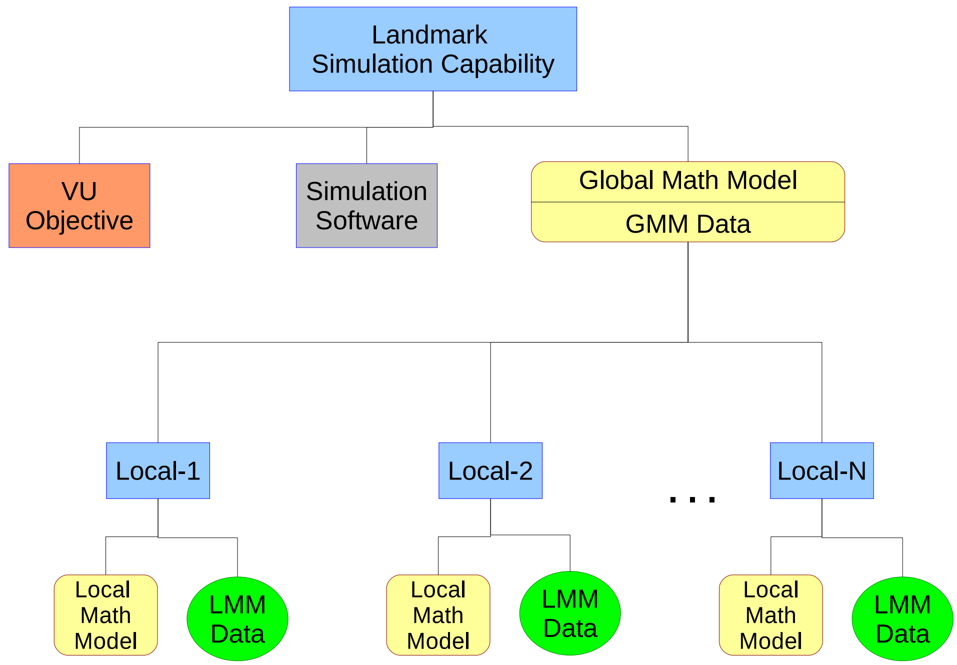

Figure 3. Conceptual Parts of a Landmark Simulation Capability

3.3 Simulations

Simulations use simulation capabilities to create output for a variety of uses. See Figure 4 for a graphic look at a simulation. Initial data is simulation software input data which can be varied to produce a variety of simulations. Raw output is the data which the software outputs as a result of a calculation. Post-processing software takes the raw output and computes processed data which can be consumed by the user of the capability. In some cases, there may not be any post-processing software (and thus processed output). Every simulation must contain initial data, a simulation capability, and raw output data.

A particular *simulation* is an application of a specific simulation capability using specific initial data. A specific simulation capability is one in which the simulation software is specific (e.g., a version number) and for which specific models and model data are used. A simulation generates specific raw output data and, with a specific version of the post-processing software, processed output data.

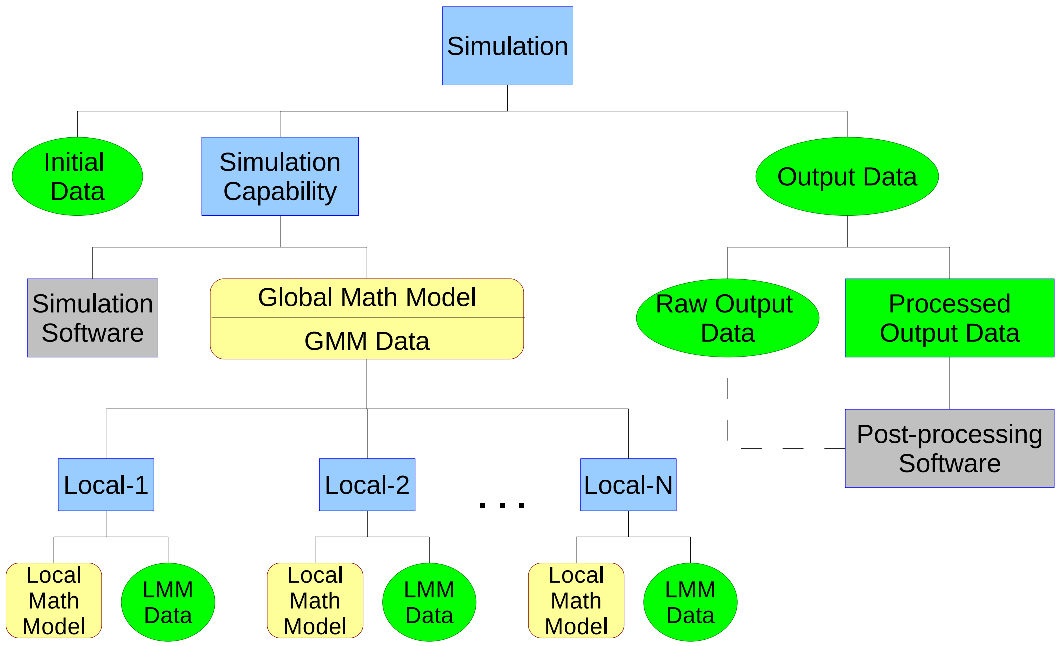

Figure 4. Conceptual Parts of a Simulation

To identify a particular simulation, one may need to identify

1. the initial data file,
2. the raw output data file
3. the name and version of any post-processing software,
4. the processed output data file,
5. the name and version of the simulation software,
6. the name and version of any local math models invoked in the simulation,
7. the associated local model data, and
8. the computing platform on which the simulation was run

Simulations play an important role in VU and it is important to establish *configuration control* over the simulations related to VU.

3.4 Simulation Ensembles

Simulations are frequently performed as a series or ensemble of simulations (see Figure 5). The top level of this hierarchy is a box representing the simulation ensemble. In the present schema, the individual simulations within an ensemble are assumed to all use the same simulation capability. Thus, within a particular simulation ensemble box we will assume that there will be a mechanism by which the specific simulation capability that was used to produce the output can be identified.[1] Below that are N boxes representing N simulations. Underneath each simulation, there is the Initial Data and the Output Data (Figure 4, but with no simulation capability). A simulation ensemble must contain at least one simulation.

Common examples of simulation ensembles within VU are mesh refinement studies, parameter sensitivity studies, and software test suites. Tracking of simulation ensembles, as well as simulations, is valuable in understanding the VU aspects of the work.

[1]The way this is actually handled is described in Chapter 6.

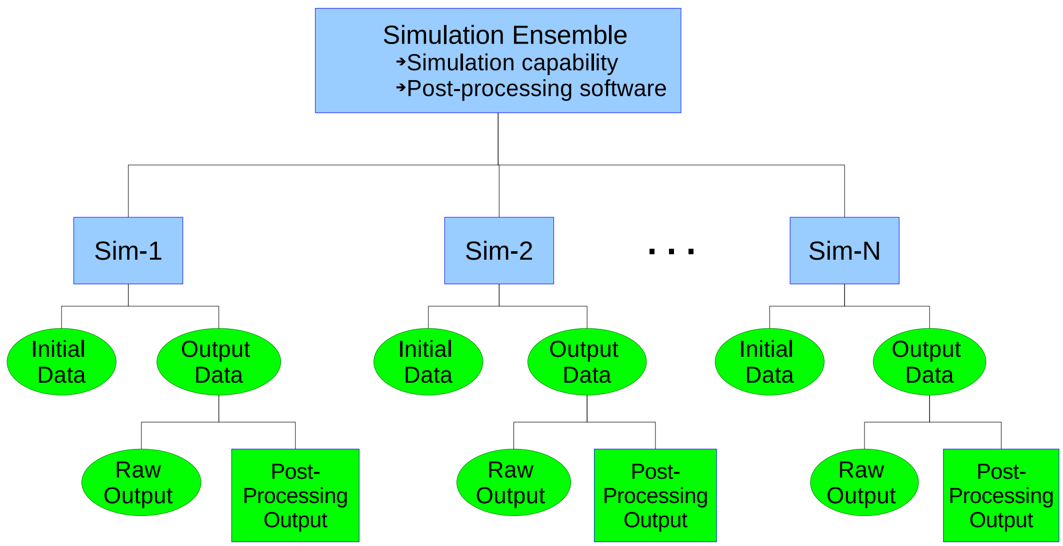

Figure 5. Simulation Ensemble

4 The ISC Hierarchy

This chapter describes the concept of an *Integrated Simulation Capability* (ISC). The ISC concept is significant to VU because it accounts for the common situation in which lower-level simulation capabilities are integrated into a single higher-level simulation capability. As the final end-product, the integrated capability can be used to simulate the response of a engineered or other system using specific models of materials, physical processes, and engineering designs. ISC's are self-contained, meaning that they require no additional simulation capabilities nor must they feed into some higher level capability in order to perform the end-use simulations. Usually ISC's can be represented as hierarchies. Please note that the hierarchies described here can be highly degenerate without breaking the overall set of processes we will describe. An ISC with only one simulation capability, for example, is allowable.

4.1 Integrated Simulation Capabilities

Before describing ISC's and their corresponding hierarchies, we need to clarify an important point. ISC's can potentially describe that which currently exists on a project or they can describe the ultimate product that the project is attempting to produce. In the Discovery process, the goal is to describe, in hierarchical terms, the final end-product, beginning with what currently exists. In so doing, the process establishes place-holders for what will ultimately exist. As VU data and evidence come into existence, the various place-holders within the hierarchy are filled with data that can be used to perform the Accumulation and Assessment processes. Any on-going VU work within the project should have a place within the established hierarchy. If it does not, then either the hierarchy is incorrect (or out of date) or the work is not relevant. At any given time, the ISC hierarchy thus can reflect the status of the VU work *within the context of the ultimate set of goals*.[2] [3]

As an illustration of this concept, we describe how the capability development work on the Waste project is organized. There are several important simulation software units within the Waste ISC[4]: (1) the Cantera library (containing solver algorithms), (2) a Cantera driver code, (3) the Yucca Mountain thermo-dynamic database, (4) a thermal-hydrology code, (5) a mechanics code, and (6) a multi-physics integration code. None of these units were developed directly by the Waste project; all are imported (or inherited) from elsewhere. The main work of the Waste project, in developing this ISC, is to augment the individual units with additional models that

[2]It is recognized that project plans change with time, so that the hierarchies may need to be updated from time to time.

[3]For mature projects a hierarchy can still be described. Because the hierarchy is not intended to be a historical record of past development, the hierarchy for a mature capability project may be rather flat, but the Discovery process can still proceed. In this case, the process will likely uncover more gaps in the evidence needed to tell the VU story.

[4]Other units also exist.

are relevant to Waste storage, and to integrate the units into a high-level Thermal-Hydrological-Chemical-Mechanical (THCM) integrated simulation capability. Even as the Waste team performs this augmentation and integration work, the software units listed above are being simultaneously developed by other teams outside the project and over which the project has little control. As new versions of all these units are created, they are imported and integrated within a separate Waste repository over which configuration control can be maintained. In order to do the integration properly, any VU-related work that is done on any of the units (either inside or outside of the project) must be both importable into the project and repeatable within the project repository as one moves up (and down) the ISC hierarchy. Therefore, the hierarchies described in this report are intended to represent the end-product as it will exist within the Waste repository. Modelers and developers are free to work with whatever codes they have at hand and in whatever order is convenient, as long as it is understood that their work on the Waste project ultimately contributes to the end-product (including its VU pedigree) as it exists within the central Waste repository.

Other projects will not necessarily strongly decouple the development work from the repository that is under configuration control and which contains the data related to VU. Whether or not there is significant decoupling within the project, the important thing with respect to our processes is that the VU work that is performed at various stages of the project can be easily integrated and reproduced within the repository as one traverses the hierarchy. The hierarchies described in this report provide a contextual framework within which this integration and re-use can be performed.

Figure 6 shows the conceptual parts of an integrated simulation capability. Capability VX is entered into the central repository and is assigned a VU objective. The Figure shows that this capability does not have a local math model. A local math model is added to capability VX to produce capability VY. A VU objective is assigned to capability VY, perhaps having something to do with the added local math model. The simulation software between VX and VY is different, if only because new lines have been added to enable the new local math model. The global math model is probably unchanged between the two versions. Meanwhile, capability VZ has been added to the repository and is assigned an appropriate VU objective. At some point later, capability VXX is created by merging capabilities VY and VZ. Merging can be accomplished in various ways such as writing a wrapper around the two lower capabilities or by stripping out items from one capability and adding them to the other. Again an appropriate VU objective is identified, for capability VXX. And so on, up the hierarchy. One could, in principle, develop the top-level capability in the hierarchy (version VZZ) and then assign a series of VU objectives to that version, rather than performing VU on each of the capabilities that are lower down in the hierarchy. Experience has shown that (and this is where requirement I(g) comes from) it is better to perform VU concurrently with the development of the individual capabilities. The simulation capabilities within an ISC are intended to represent *landmark* simulation capabilities of importance to VU. The hierarchy is best understood as corresponding to an idealized simulation capability 'development sequence', based on the idea of 'landmark' simulation capabilities.

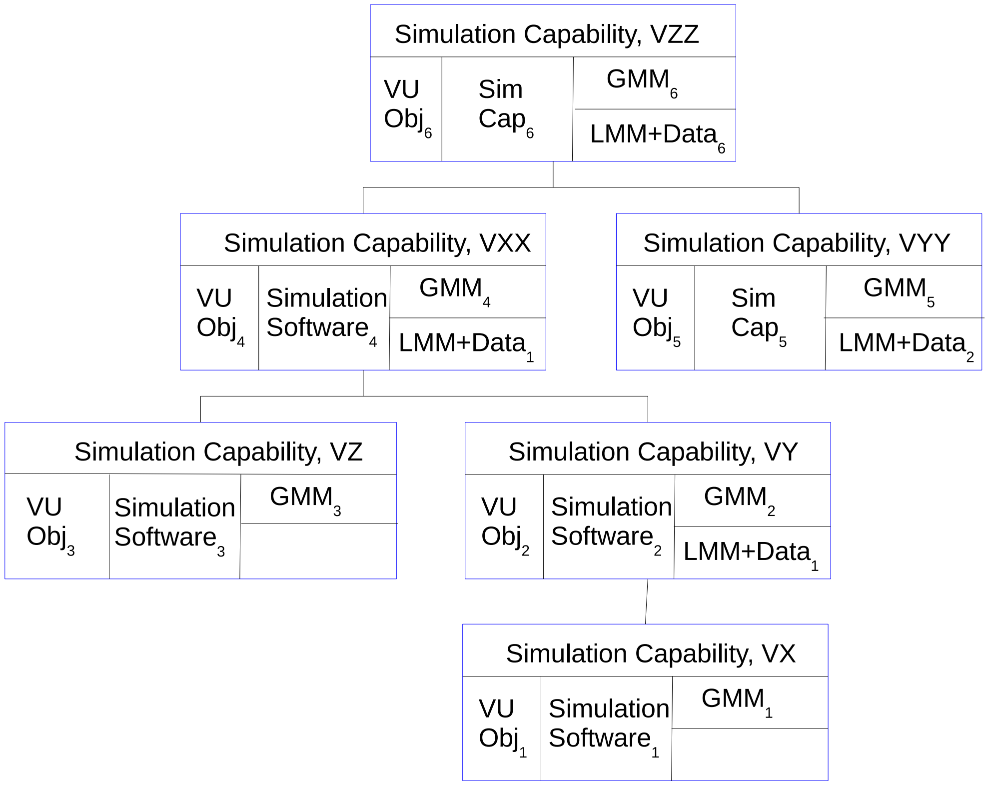

Figure 6. Conceptual Parts of an ISC Hierarchy

4.2 Example: The Waste ISC Hierarchy

As an example of an ISC, we describe an ISC under development within the NEAMS Waste project. The ISC corresponds to an integrated suite of thermal, hydrological, chemical, and mechanical (THCM) simulation capabilities.

The hierarchy is best understood as corresponding to a simulation capability 'development sequence', based on the idea of 'landmark' simulation capabilities. The THCM development sequence is described next, with each item in the list representing a landmark simulation capability. Note that for conceptual purposes the landmark simulation capabilities are given version numbers VA, VB, VC, etc. As development proceeds the actual software version numbers become known. It is important to understand that this capability development sequence is simply a logical ordering of the development activities which is used to provide a hierarchy upon which an assessment can be based. It is not necessarily the order in which capabilities will actually be developed.

1. Glass-Brine Dissolution (Chemical Equilibrium Calculation), version VA. This is the version of the simulation capability that is acquired and integrated into the official project repository known as the SMP.

2. Glass-Brine Dissolution (Chemical Equilibrium Calculation), version VB. This is the version of the capability that performs chemical equilibrium calculations for a specific brine and glass mixture. It is a landmark version because the glass/brine dissolution model is validated.

3. Glass-Brine Dissolution (Chemical Equilibrium/Kinetics), version VC. This is the version of the capability that incorporates chemical kinetics. It is a landmark because chemical kinetics (a new model) is added.

4. Thermal-Hydraulic capability, version VA. This is the version of the Aria capability that is acquired and integrated into the SMP. It is a landmark version because it is an inherited capability.

5. Thermal-Hydraulic-Chemical capability, version VB. This is a landmark version of the capability that has integrated the chemical equilibrium model (Glass-Brine Dissolution, version VC).

6. Mechanical capability, version VA. This is the version of the Adagio capability that is acquired and integrated into the SMP. It is a landmark version because it is an inherited capability.

7. Mechanical, version VB. This is the version of the capability that contains the salt creep model. It is a landmark because a new model was added.

8. Multi-physics Coupling capability, version VA. This is the version of the capability that is acquired and integrated into the SMP. It is a landmark version because it is an inherited capability.

9. THCM, version VB. This is the landmark version of the THCM capability that has integrated the THC capability , version VB with Mechanical, version VB. The coupling is one-way.

10. THCM, version VC. This landmark version of the THCM capability has two-way coupling.

11. Dakota, version VA. This is the version of the software that is acquired and integrated into the SMP. It is a landmark version because it is an inherited capability.

12. The stochastic PA software, version VA. This software uses Dakota and THCM, version VC to do UQ in an end-use setting. It is a landmark because this is the top-level simulation capability.

Multiple other versions of the software may exist between the landmark versions if there is no VU activity associated with it.

Development within the ISC may continue, for example, by adding a new model to Arpegio (producing version D) to represent a new waste form, barrier system, or local host medium.

The software development sequence above may be represented by a hierarchy of landmark simulation capabilities (see Figure 7).

Note that the order in which these different capabilities can be developed must proceed logically so that integrated capabilities require the development of the lower-level capabilities beforehand.[5] The logical ordering is not necessarily unique and some work can take place in parallel.

In closing, a word is necessary on how this capability development sequence was 'discovered'. The Waste project created several documents related to PIRT and GAP analyses in which the THCM goal is described. Certain steps within the process, such as adding a glass dissolution model to the chemical equilibrium code were characterized in the documents as Challenge Problems. From these descriptions we constructed an initial guess as to what a logical development sequence might be for the THCM capability. Iteration with the development team resulted in Figure 7. Since much of the development work was only beginning, the items in the upper part of the hierarchy were somewhat speculative. It may be possible, in the future, to devise a questionnaire to be given to the developers in order to speed up this part of the Discovery process (see Chapter 7 for a discussion of the VU questionnaires used in this process).

[5] In practice, the time at which a particular component capability is integrated into a higher-level capability may be somewhat flexible, so that both capabilities are being developed at the same time.

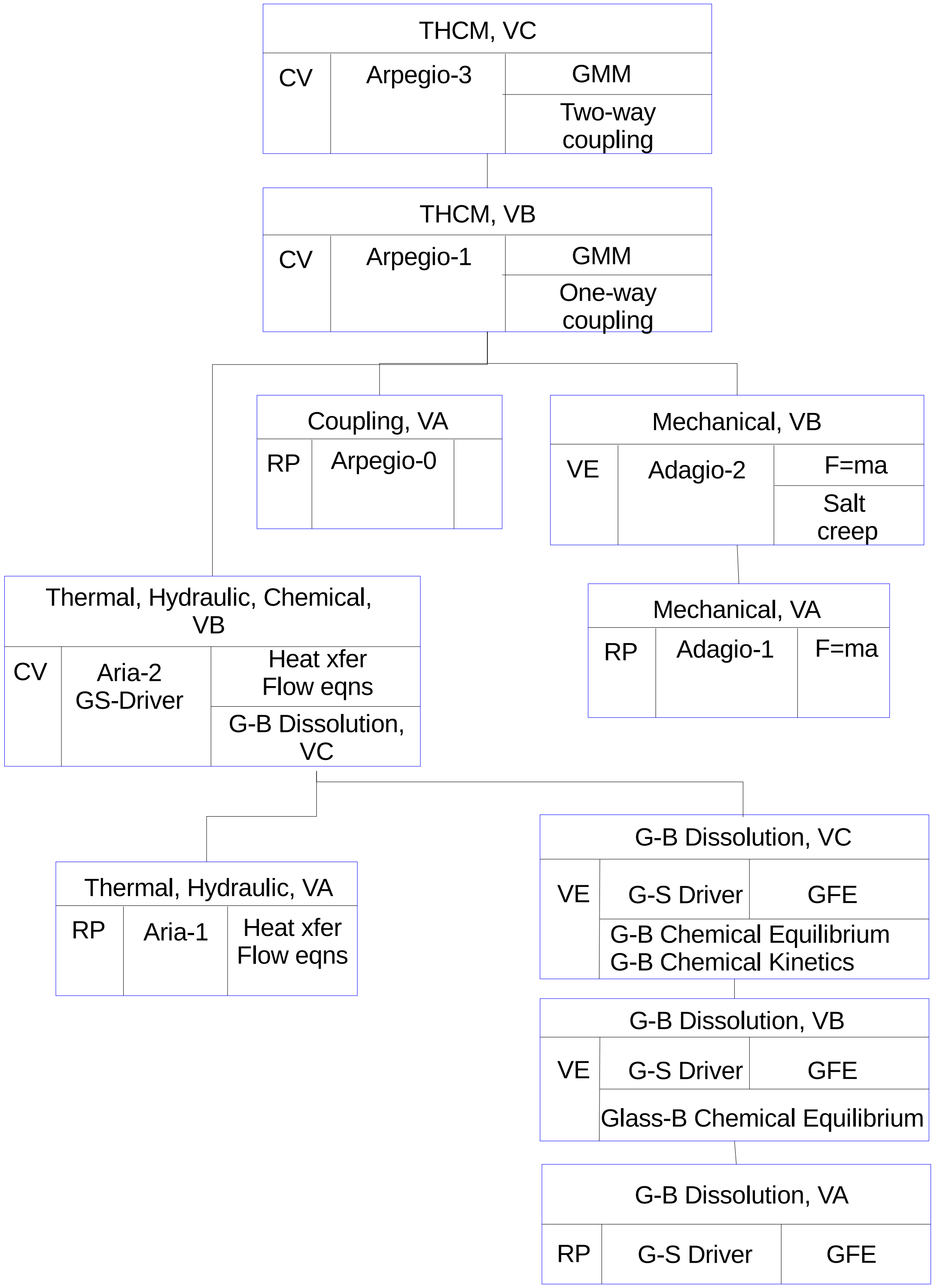

Figure 7. The Waste THCM ISC Hierarchy

5 Discovery: Particularizing the ISC Hierarchy

The last two chapters described abstract hierarchies pertaining to simulation capabilities, landmark simulation capabilities, simulations, simulation ensembles, and integrated simulation capabilities. Each of these hierarchies needs to be particularized when applying them to a specific large simulation capability development project. A preliminary step in particularization of the hierarchies is to discover the names of the entities represented by the boxes in the specific hierarchy and, in so doing, determining how many boxes are within the specific hierarchy. For example, Figure 1 shows the abstract simulation capability hierarchy, while Figure 2 shows a specific simulation capability. Notice that the number of boxes in each of these hierarchies is not the same, nor should they be. Another example of preliminary particularization relates to the examples of the abstract and specific integrated simulation capability hierarchies shown in Figures 6 and 7. In creating the specific ISC hierarchy for Waste, we relied on the concept of a landmark simulation capability development sequence, as described in Section 4.2.

5.1 Formalization of the Hierarchical Description

The hierarchies of entities (or boxes) that we've described make clear the relationships between different kinds of things that go together. For example, the input and output of a particular simulation go together because one produces the other, with the help of a simulation capability. The hierarchies we've been discussing are not classification systems, they are just collections of things that are related to one another in a particular fashion best described by a hierarchy. The main reason we need hierarchies is to establish context. For example, when one considers code verification in relation to a specific code (or code version), it suggests that all parts of the code should be verified. On the other hand, if one considers code verification within a VU hierarchy, it suggests that only those parts of the code that are used in performing the simulations within the hierarchy need to be verified.

Each of the hierarchies we've described is a collection of 'boxes' or entities. For example, the simulation capability hierarchy contains a box representing itself at the top, and other boxes below representing simulation software, math models, and associated data. The boxes within the hierarchy can represent equivalent things. For example, in the simulation ensemble hierarchy there are multiple boxes, each representing a simulation. In this case, the boxes correspond to similar things. In general, however, the hierarchies we need contain a mixture of conceptually different things. For example, in our schema, the math model is not equivalent to it's implementation in software and thus there are different boxes for the math models and the software.

Although the ISC and its simulation capabilities potentially contain many entities, the number of entity *types* is considerably less. For example, there is a simulation entity type, a local math model entity-type, a simulation software entity-type, and numerous others. The complete list of entity types which have been identified in the ISC hierarchy we've described shows that there are 7 entity types associated with the ISC hierarchy:

- Landmark Simulation Capability,
- Simulation Software,
- Global Math Model,
- Global Math Model Data,
- Local Math Model,
- Local Math Model Data, and
- VU Objective.

Notice that there is no ISC entity-type because the top-level entity in the hierarchy is itself a landmark simulation capability.

There are a number of variables common to all entity-types within the landmark simulation capability hierarchy and the other hierarchies, to which one can assign values in order to create the entity as a place-holder within the hierarchy. We shall call these variables the *essential* variables. This data is used to name the entity and to describe it's position within the hierarchy:

1. entity name
2. entity version (an ID to indicate versions within a sequence of versions of the entity)
3. entity date (date version was created)
4. pointer to parent entity (usually one parent unless its the top-level entity)
5. number of child entities
6. pointers to child entities
7. pointer to location of item itself[6]

[6] As an example of this, consider a particularized simulation capability. Not only is the software within named and version-ed, one can also include some sort of pointer which tells one where to find the actual source code within the central repository.

Except for the last item, concrete data must be assigned to all of the essential variables in order to define an entity within the hierarchy. To particularize a hierarchy (like a simulation capability), we must particularize each of the non-optional entities it contains.

Formally, the first part of the Discovery process is a matter of particularizing the essential variables describing the hierarchies. The essential variables, when particularized, can be used to traverse the hierarchies that have been defined to obtain contextual information. Using Figure 3 as an example, one could begin within one of the green entities representing the data associated with a particular local math model and, by traversing the hierarchy along various paths, find its associated local math model, the global math model to which it belongs, the simulation software which uses the data, and the VU objective(s). Conversely, beginning with the VU objective, one should be able to trace back to the specific local math model at which the objective is aimed (this will require an auxiliary variable).

Every entity-type and every particular entity contains a *location* pointer (item 7 in the list above). Let us consider the meaning of this pointer in relation to the entity-types. For most of the entity-types, the pointer is either a pointer to a root directory within the SMP or it is a pointer to a specific critical file. For example, the simulation capability location pointer points to the root directory where one can file the files associated with the particular landmark simulation capability in question. The simulation software pointer points to the makefile (or to the 'main') corresponding to the software associated with the particular simulation capability. The global math model pointer points to a file which contains a description of the global math model (this is most likely some sort of text document containing either the equations themselves or pointers to documents from the literature). The default setting for each location pointer is NULL. By this means, one can determine whether or not the place-holder has been filled. The VU objective pointer points to a root directory or perhaps to a short document which explains why this objective is appropriate.

Some entity-types may contain additional *auxiliary* variables that are unique to the entity-type. These variables are the distinguishing feature between different entity-types. Every entity of the same entity-type will contain the same auxiliary variables. As an example of an auxiliary variable, the entity corresponding to a simulation ensemble should contain an identifier for the computer platform on which the simulations were run. The auxiliary variables need not have concrete data assigned to them in order to particularize an entity. If all the essential and all the auxiliary variables have concrete data assigned to them, then the entity is fully particularized. Only the essential variables are needed to particularize a hierarchy, so we will defer definition of these auxiliary variables to a later chapter. The function of the auxiliary variables is to provide place-holders for the information and evidence that is needed to tell the VU story.

The second part of the Discovery process is to particularize the auxiliary variables for each of the entities in the hierarchy. This will be addressed in Chapter 8.

5.2 Example: Particularizing the Waste THCM ISC Hierarchy

The first step in the Discovery process is to identify the ISC's within the project. As Figure 8 shows, a large project may possibly develop more than one ISC. If that is the case, then one must define each of the ISC's, one at a time. For the case of Waste, there is currently only one ISC hierarchy, namely the THCM hierarchy.

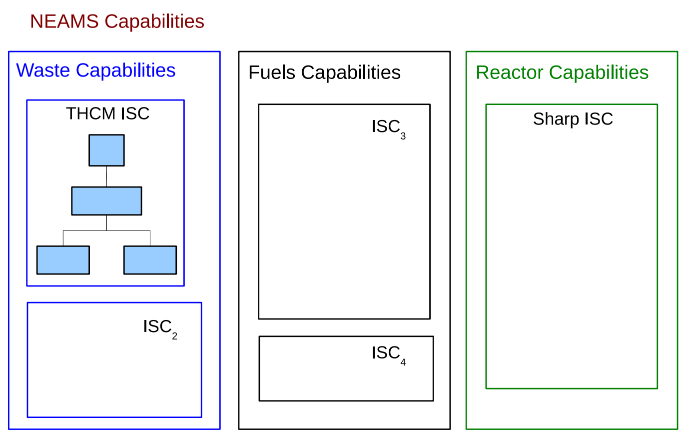

Figure 8. Integrated Simulation Capabilities within the NEAMS Context

The next part of the Discovery process is to define each individual ISC hierarchy. The hierarchy is created by first establishing the landmark simulation capability development sequence. This involves not only listing the simulation capabilities as was done in Section 4.2, but by particularizing the simulation capability hierarchies for each landmark simulation capability. For example, the first landmark in the development sequence was Glass-Brine Dissolution (Chemical Equilibrium Calculation), version VA. Therefore, one creates the simulation capability hierarchy shown in Figure 9. Note that this figure is the same as Figure 2, except that, because it is the hierarchy for a landmark simulation capability, a VU objective has been added. The VU objective in this example is RP (research pedigree) since it is a 'completed' capability. The next step in the development sequence is to develop the simulation capability Chemical Equilibrium Calculation for Glass-Brine Dissolution, version VB. One should create the simulation capability hierarchy for version VB. This hierarchy should be similar to that for version VA, but the version VB hierarchy must reflect some change that was made to the capability. In this example, the VU objective was changed from RP to VE since there is experimental data against which the model can be validated. It may also be the case that some of the particular version numbers within the entities have changed (e.g., a different version of the Cantera Library is used). In that case, version VB would have two VU objectives: VE (for the model) and RP (for the change to the library). One continues on through the development sequence until all the landmark simulation capability hierarchies have been defined. Having the development sequence in hand, one can draw the ISC hierarchy.

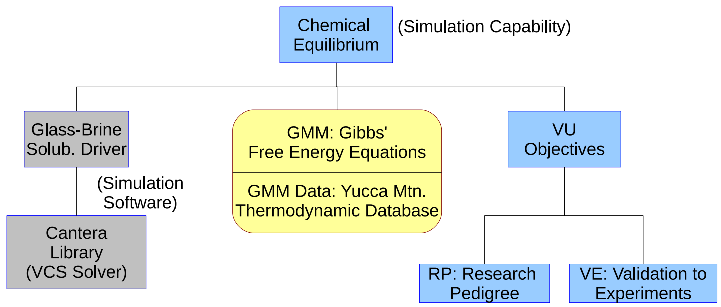

Figure 9. The Landmark Glass-Brine Chemical Equilibrium Capability

The Discovery process is best performed at the time the project or ISC work is initiated because it could potentially impact project plans. The process may take some time to complete and will require participation by a number of members of the project team. Within the Waste Project, the data describing the hierarchy and development sequence will be entered into the knowledge management system using the Velo software. Collecting the proper information may take a series of meetings or perhaps interviews with key project members. Some parts of the hierarchy may need to be defined or updated later as additional information becomes available.

5.3 Completion of the ISC Hierarchy Identification Process

After all the essential variables have been defined, the ISC Hierarchy has been particularized. To complete this process, we need to verify its fidelity to the long-term modeling and development plan within the project. This will be accomplished by a review by the appropriate development team members. To facilitate the review, figures displaying the various parts of the hierarchies should be generated from the essential variables stored in the Evidence Management System. The primary purpose of the review is to ensure the information is complete and correct, and that there is basic agreement among team members on the description of the landmark capability development sequence. Finally, an analysis should be under-taken by the review team of the hierarchy of VU objectives defined by the ISC to ensure coherence and consistency (in the sense of a Validation Hierarchy). The result of the review will be a preliminary *ISC Narrative* which describes the development sequence in terms of landmark simulation capabilities and a discussion of VU objective coherence.

This process is only the first part of the Discovery process. There remain additional steps involving identification of the VU hierarchies within the ISC and determining the auxiliary variables for both the ISC and VU hierarchies.

6 VU Hierarchies

The hierarchies in the previous chapter play an important role in VU. Before that role can be described, however, we need to describe another hierarchy, called the *VU Evidence Hierarchy* (see Figure 10). The top entity in the VU Evidence Hierarchy is the VU objective. The child entities are the VU elements within the element-chain corresponding to the VU objective type. Below each VU element are five simulation-based practice areas described in section 6.4. The third simulation practice area pertains to a simulation ensemble. The entities under the simulation ensemble are the same as in Figure 5. The entities within this hierarchy are place-holders for organizing the VU aspects of the ISC. As noted previously, the VU objective at the top of the Evidence Hierarchy also belongs to the Landmark Simulation Capability hierarchy. In this way, VU is coupled to specific simulation capabilities, and specific simulation capabilities have an associated VU pedigree established by the activities and results of the Evidence Hierarchy. Details on the entities within the VU hierarchy are given in the remainder of this chapter.

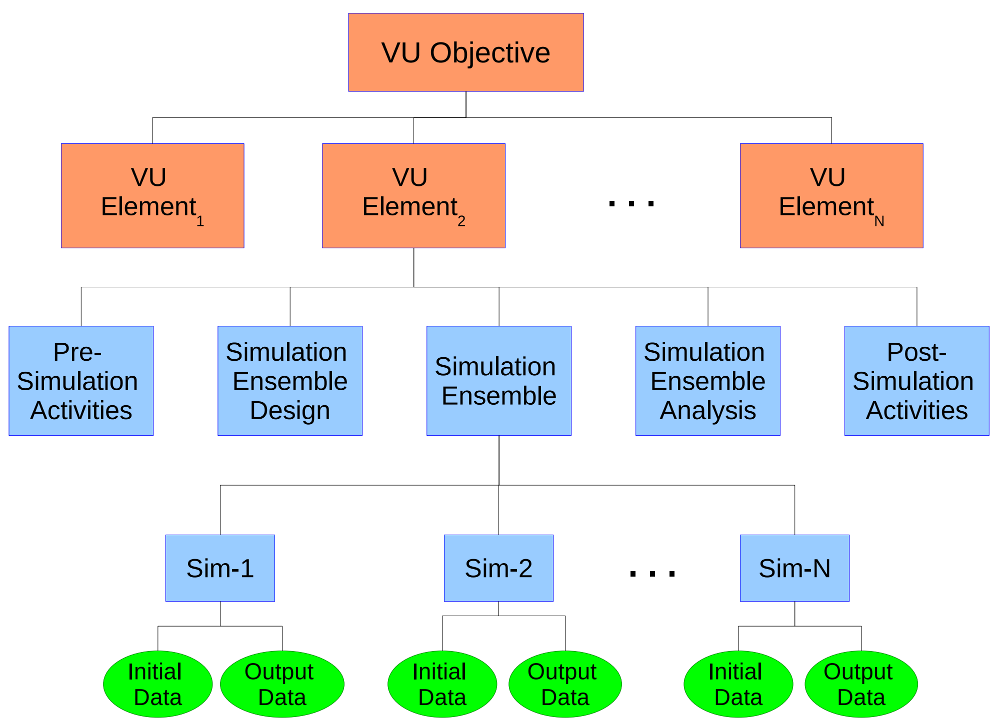

Figure 10. Conceptual Parts of the VU Evidence Hierarchy

6.1 VU Objectives

A *VU objective* is associated with the development or use of a particular landmark simulation capability. The objective determines the goal of the VU work that is associated with the particular simulation capability (or a part thereof). VU objectives are essentially 'use-cases' of VU. Because simulation capabilities may contain multiple local models, there can be more than one VU objective associated with a given simulation capability. Each of the individual objectives would have its own VU evidence hierarchy underneath. For clarity in discussing VU objectives below, we need to distinguish between the simulation capability to which an objective is being assigned and other simulation capabilities that may play a role within the hierarchy. Consistent with our prior terminology, let us call the former the *landmark* simulation capability and the others *auxiliary* simulation capabilities. VU objectives may be assigned to auxiliary simulation capabilities, but these must be done within the context of the hierarchy pertaining to the landmark capability.

We list a number of common VU objectives that could be assigned to a particular simulation capability.

VE *Validation to Experiment.*
The objective is to validate one of the local models within the landmark simulation capability against available experimental data.

CD *Performing Simulations to Create Data.*
The objective is to use the output of a simulation capability to create data that is associated with a local model within the landmark simulation capability. The output is usually produced using an auxiliary simulation capability. A simulation capability with CD as its goal will typically augment or replace existing data associated with one or more of its models.

CLM *Performing Simulations to Create Local Models.*
The objective is to use the output of an auxiliary simulation capability to help create a local math model to be used within the landmark simulation capability. A major example is the use of a sub-continuum code to create a local model which will be used within a high-fidelity code. A simulation capability with CLM as its goal will typically create a local math model which augments the list of local models within the landmark simulation capability.

CGM *Performing Simulations to Create Simplified Global Models.*
The objective is to use a simulation capability to create a simplified global model within the landmark simulation capability. A major example is the use of a high-fidelity simulation capability within a UQ capability code to create a surrogate model. The landmark simulation capability would then consist of a driver code which samples the surrogate model.

EU *Performing Simulations for End-Use Analysis.*
The objective is to use an integrated simulation capability to make a prediction for an end-use application.

VS *Verification of Software.*
The objective is to verify relevant portions of the simulation software within the landmark simulation capability.[7] A simulation capability with VS as its goal will typically verify only a subset of all the features and capabilities of within the simulation capability. These F/C's would be those with relevance to the VU objective assigned to the capability.

RP *Research Pedigree for Inherited or Completed Capabilities.*
It is not uncommon for data, local or global models, simulation software and/or entire simulation capabilities to be inherited from sources outside the control of the project. Inherited capabilities may, in some cases, have their own VU pedigree which, if sufficiently documented, can be used to establish their pedigree either within the landmark simulation capability or relative to their use as an auxiliary simulation capability. If the documentation is poor, one may have to do some type of archival or folkloric research to learn what, if anything, was done in terms of VU. The other situation in which this objective applies is when a capability within the project is largely completed before the data pertaining to it has been identified by the Discovery Process and/or stored within the Evidence Management System.

Compound VU goals can also exist and form a sequence (e.g., VE, then EU).

6.2 The Elements of VU

Practice or topical areas are often used to describe the heterogeneous collection of activities which comprise VU. For example, [3] identified 12 topical areas related to VU. The Predictive Capability Maturity Model lists six top-level practice areas, sometimes referred to as *VU Elements*. In this document we adopt the VU elements from the PCMM as the basis for our VU hierarchy.

Elements, as defined by the PCMM, are practices associated with modeling, verification, validation, and uncertainty quantification. Characterizations of these elements are given below, with detailed descriptions found in [3], [6], and other documents.

According to [6], the top-Level Elements of VU are:[8]

[7] For clarity, we use VS for the objective and CV for the VU element.

[8] We have taken a few liberties with the abbreviations of the elements and re-named the original 'Physics and Material Model Fidelity (PMMF)' element to Math Model Development. What is not changed is the content and scope of the original elements.

E-1.0 *Representation and Geometric Fidelity (RGF)*. Having to do with the geometry or other representation of the true physical system vs. the geometry of the computational representation of the system.

E-2.0 *Math Model Development (MMD)*. Having to do with the mathematical representation of the physical system, along with its parameters.

E-3.0 *Code Verification (CV)* Having to do with the correct implementation of the mathematical model in software.

E-4.0 *Solution Verification (SV)*. Having to do with the integrity of the input and output of the computer model when the latter is used in validation or to make a prediction.

E-5.0 *Local Math Model Validation (MMV)*. Having to do with comparison of computer model output (predictions based on a particular local math model) and experimental results.

E-6.0 *Uncertainty Quantification (UQ)*. Having to do with sensitivity analyses, uncertainty propagation, and margins.

The VU elements listed above are an important part of the VU hierarchy.

The elements within PCMM are further sub-divided into sub-Elements addressing a variety of heterogeneous topics within each VU element. The sub-Elements do not appear explicitly in our VU hierarchy, but are incorporated into the Discovery process in two different ways. The first is via the questionnaires in Appendix I; these will be discussed in the next chapter. The second is through the use of the Simulation Ensemble concept, which we re-visit in section 6.4. First, however, the next section introduces the idea of an Element-Chain.

6.3 VU Element-Chains

The VU Elements described in the previous section should not, for the most part, be performed in isolation. Rather, they are properly performed in a logical sequence which we refer to as an Element-Chain. An Element-Chain is a sequence of VU elements whose ordering forms a logical sequence of activities or practices. In our VU evidence hierarchy, each VU objective has a pre-defined Element Chain. In this way, the chain serves as the bridge between an objective and the relevant VU elements, within a specific VU hierarchy. Two well-established VU element-chains are identified in this section.

The first well-established element chain is associated with the VE objective. Figure 11 shows the element chain associated with the VE objective. The basic logic of the VE chain is to first assess the fidelity of the representation of the conceptual model and also the geometry of the experimental set up. Next one assess the degree of due diligence that has been applied to the math model and its associated data. Then, before one can make predictions using the simulation capability, one must verify the correctness of the simulation software (code verification element). In making the predictions, one verifies the input and output of the simulation ensemble, assesses the sensitivity of the solution of mesh and other numerical parameters, and estimates discretization error. Once the predictions have been made, uncertainty is quantified. In the Validation process (MMV) the predictions are compared to the experimental data (ideally using validation metrics). A conclusion is reached in terms of the range of validity of the model.

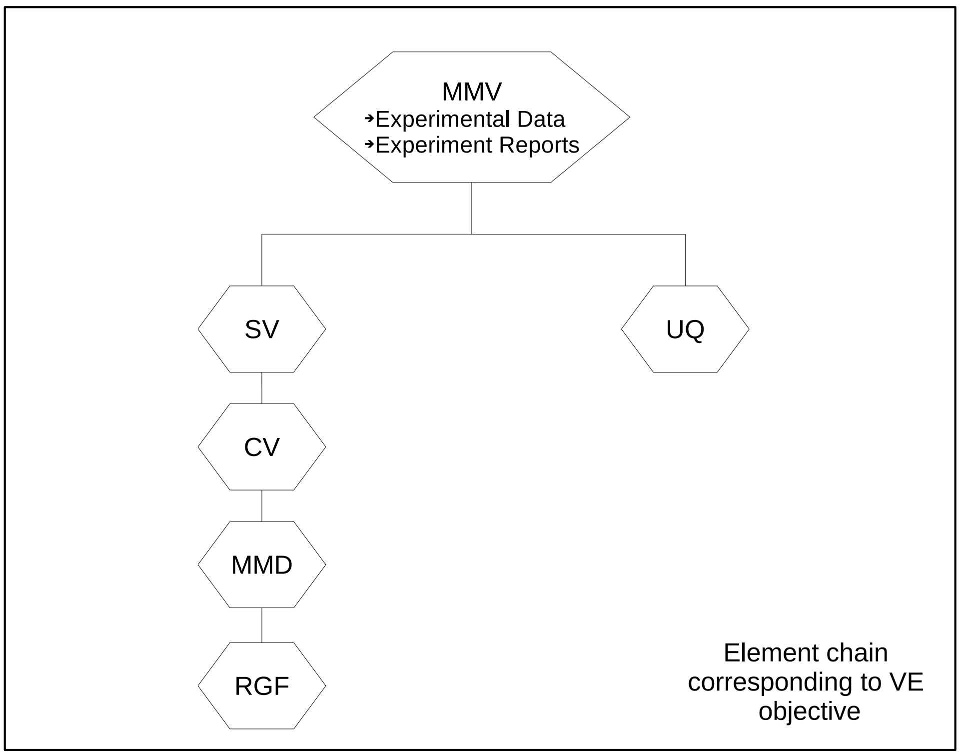

Figure 11. VU Element Chain associated with the VE Objective

The second well-established element chain is associated with the EU objective. Figure 12 shows the element chain associated with the EU objective. In this sequence of elements, one begins by considering the extrapolation issue: how does the end-use parameter range for the math models compare to the parameter range under which the model was validated? Next, are there any features and capabilities within the simulation software that will be exercised by the end-use simulations that have not previously been verified? Then, the end-use predictions (solutions) must be verified

(SV hexagon), and UQ analyses performed. In the EUA hexagon (End-use analysis) one presents the analysis results in a form which can be readily consumed by the end-user. Usually, the EU element chain is preceded by a VE or other element chain which established the VU pedigree of the simulation capability.

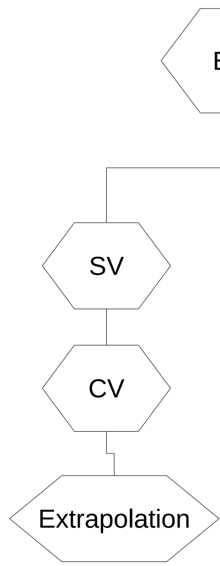

Figure 12. Logical Ordering of VU Elements within the EU VU Objective

A third (trivial) chain is associated with the VS objective. In this case, the VU Element Chain consists of only one element, namely CV.

Our collaboration with Waste identified several other VU objectives for which we are uncertain as to the corresponding VU Element Chain. Future collaborations will help sort out this issue.

6.4 VU Simulation Ensembles

VU-simulation ensembles are simulation ensembles (as defined in Section 3.4) whose purpose is tied to a specific VU element from the list above. The following types of VU-simulation ensembles exist:

1. *Calibrations.* Within Math Model Development, simulations performed to calibrate a model against selected data are considered VU simulation ensembles. Any sensitivity studies performed in connection with Calibration are also VU simulation ensembles.

2. *Verification Tests.* Within Code Verification, simulations performed to verify the relevant features and capabilities of the simulation software within the context of the VU objective are considered VU-simulation ensembles. They do not automatically include everything that is found within the typical the regression test suite. Later regression testing against the gold copy test results associated with this simulation ensemble is essential, however.

3. *Numerical Model Sensitivity Studies.* Within Solution Verification, examples of VU simulation ensembles include mesh refinement studies to establish convergence behavior or studies that explore the sensitivity of the numerical solution to numerical model parameters such as artificial damping or solver tolerances.

4. *Making the End-Use Predictions.* Within Uncertainty Quantification, simulations that are performed to make the end-use prediction, are examples of VU simulations.

5. *Sensitivity Studies.* Within Uncertainty Quantification, simulations that are performed by sampling physical or geometric model parameters are VU simulations.

6. *Simulations that Feed into Model and Data Development.* Within Math Model Development or Representation and Geometric Fidelity, simulations performed to assist in the development of a model or to create math model data are considered VU simulation ensembles.

7. *Simulations that Create Surrogate Models.* Under Uncertainty Quantification, simulations or simulation ensembles that create surrogate models for performance assessment may be considered simulation ensembles.

Other examples of VU simulation ensembles can probably be given.

VU simulation ensembles are incorporated into the VU Hierarchy via a set of five entities having to do with simulation practices. The set of simulation practice entities can occur under any of the six VU Elements.

S-X.1 *Pre-simulation Practices.* VU practices that are not specifically tied to a simulation ensemble, but which lay the groundwork to enable simulation or generate other kinds of evidence.

S-X.2 *Simulation Ensemble Design.* For a given VU element, design an appropriate and particular simulation ensemble. Further, define each simulation precisely in terms of the items which make up a simulation (as outlined in section 3.3) so that the simulations within VU can be put under configuration control and are assigned their proper place in the hierarchy.

S-X.3 *The Simulation Ensemble.* Verify the code inputs, and verify code outputs.

S-X.4 *Simulation Ensemble Results Analysis.* Create plots, figures, tables, explanations, reports, and conclusions based on the simulation ensemble output.

S-X.5 *Post-simulation Practices.* VU practices that are not specifically tied to a simulation ensemble, but which depend upon the simulation analysis.

Notice that the five VU simulation practice areas defined above occur in a logical order so that S-X.1 is performed first, S-X.2 is performed second, and so on. The S in these designations stands for 'simulation' and the X in these designations refers to which of the VU elements the simulation practice area is attached. For example, S-2.3 refers to applying good simulation practices with respect to the VU simulation ensembles under the Math Model Development element (E-2.0).

Some of the original sub-Elements identified within the PCMM are strongly related to these simulation practices, while others are not. For example, sub-Element UQ2 (Sensitivity Analysis) clearly involves simulation ensembles that can be placed within the context of the hierarchy. Some of the original PCMM sub-Elements may not be not covered by this simulation-based hierarchy, but it is expected that they can be incorporated elsewhere within the system of processes that will eventually be defined. The inclusion of simulation practice entities within the hierarchy provides context for the many simulations which should be performed in order to create a proper VU pedigree. Moreover, they create place-holders for some activities which are not called out by the PCMM and additionally allow for configuration control of important simulations.

7 Discovery: Particularizing the VU Evidence Hierarchy

As explained in Chapters 4 and 5, the ISC Hierarchy is defined using the concept of the landmark simulation capability sequence. The sequence itself is discovered through meetings with the modeling and development team and/or existing project documents. From the sequence, one particularizes the ISC hierarchy by assigning values to the essential variables, providing context and allowing one to traverse the hierarchy. In this chapter we describe how the abstract VU hierarchy is particularized, using the VE objective assigned to the Glass-Brine chemical equilibrium capability (Figure 9) as an example (Section 7.1). In addition, we lay the groundwork for defining the auxiliary variables within each entity-type. The auxiliary variables create important place-holders for evidence and other important artifacts that will be needed in the Accumulation and Assessment processes that occur after Discovery. Our approach to understanding what sorts of auxiliary variables are needed is to generalize from the particular to the abstract, using the Glass-Brine example from Waste as the particular. Therefore, in Section 7.2 we present the particular, while in the next chapter we perform the generalization.

7.1 The VU Evidence Hierarchy: Particularization and Discovery

The VU Evidence Hierarchy for the Glass-Brine capability is defined by assigning values for the set of seven essential variables listed in Section 5.1, requiring one set per entity in the hierarchy. As before, we begin by listing the VU Evidence hierarchy entity types, of which there are 17:

- VU Objective,
- VU Elements (6),
- Pre-Simulation,
- Simulation Design,
- Simulation Analysis,
- Post-Simulation,
- Simulation Ensemble,[9]
- Simulation,

[9]Simulation Ensembles can exist without reference to VU, of course, but such ensembles are not important for our purpose.

- Input,
- Raw Output,
- Post-processing Software, and
- Processed Output

The VU objective entity-type at the top of the hierarchy also appears as an entity-type within the hierarchy for a particular landmark simulation capability. In terms of the first essential variable (entity name) we must assign one of seven allowable values, namely 'VE', 'CD', 'CLM', 'CGM', 'EU', 'VS', and 'RP', corresponding to the seven VU objectives described in Section 6.1.[10] For the Glass-Brine Dissolution capability, the entity name is assigned the value VE (Validation to Experiment). A box version number (e.g., version 1.0) is assigned upon creation of the box within the hierarchy. The parent box is the particular landmark simulation capability with which the objective is concerned. In our example, the parent is the Glass-Brine Chemical Equilibrium capability.

With the VU objective assigned, the number and names of the child entities within the hierarchy are automatically known.[11] As described in the previous chapter, the VE objective is associated with the VU Element Chain shown in Figure 11. Thus the number of child entities in the VE objective is six, corresponding to the elements RGF, MMD, CV, SV, UQ, and MMV. Pointers to the location of these element entities within the Evidence Management System are assigned, along with the pointer to the location of the VE objective entity itself. With that, all the essential variables for this objective have been assigned.

The next level in the VU Evidence Hierarchy concerns the six element entities corresponding to the VE objective. To define the essential variables within an element entity, we first assign the entity name. The name can be as simple as 'CV' if it is a code verification entity, or one can perhaps assign something more description like 'CV for VE of Glass-Brine'. The parent entity for the elements is clearly the VU objective entity, namely VE. Usually, there are five child entities for each of the elements. The child entities correspond to the five simulation entities defined in Section 6.4. These are named Pre-Sim, Sim-Design, Sim-Ensemble, Sim-Analysis,and Post-Sim. Pointers to the locations of all these entities are assigned to complete the definition of the essential variables.

The entities in the lower part of the Evidence Hierarchy which belong to the Simulation Ensemble hierarchy are usually not defined as part of the Discovery Process

[10]The VU Objective entity-type is the only entity-type for which the assigned entity name is restricted to a limited set of values. In most other cases, the entity name is unrestricted. This restriction can be avoided if an additional auxiliary variable is added to the VU objective entity-type.

[11]Presently this statement is only true for the 'VE', 'EU', and 'VS' objectives; eventually we hope that the Element Chain for the remaining objective types will be determined.

because to do so requires completion of the Simulation Design activity, which can potentially occur during the Accumulation process. However, the simulations can be designed before or during the Discovery process also. Overall, the assignment of values to the essential variables within the first three levels of the Evidence Hierarchy is nearly automatic, in contrast to the assignments made to define a particular ISC hierarchy.

With the particular VU Evidence Hierarchy defined, we turn to the more difficult task of defining the auxiliary variables within the individual entity boxes. Basically, these auxiliary (or place-holder) variables describe the VU aspects of the modeling and development work related to the specific simulation capability. While this task is much more difficult to automate, the specific hierarchies that have been established to this point make it somewhat easier because the context has been determined, and one can proceed systematically through the entities. To assist with this phase of the Discovery process we provide in the Appendix a series of questionnaires. The questionnaires are similar in spirit to those associated with the PCMM in that the intent is to provide a detailed, but global view of the key issues in VU. The questionnaires were used in this work as an aid to interview the modelers and developers with respect to the Glass-Brine chemical equilibrium capability. Longer-term, as the specific questions on the questionnaires become more refined, they may be provided in an on-line format. The questionnaires are divided into three sets. The first set of questionnaires (Appendix I) pertain to the six VU Elements, and thus there are six questionnaires in the set. The second set (Appendix II) pertain to the various sub-Elements under each of the six Elements, as motivated by the PCMM. The third set (Appendix III) pertain to the five boxes on the Simulation level of the VU hierarchy.

Even though we have described the Discovery processes for the ISC and for the VU hierarchies, it is not meant to suggest that in practice the latter must follow the former. It may be more helpful to iterate between the two discovery processes. In any case, discovering the VU Evidence Hierarchy should not be delayed too long after the ISC hierarchy has been Discovered. With the place-holders established, the Discovery process is nearly complete. To complete the process, a review of the VU Evidence hierarchy, including the identified place-holders, should be conducted by appropriate team members. The primary purpose of the review is to ensure the information is complete and correct, and to ensure there is basic agreement among team members on the approach suggested by the answers to the questionnaires.

One important issue not discussed yet concerning the Discovery process is, who within the project will do the discovering? Although Discovery requires a team effort, it seems necessary given the current state of VU practice within large simulation projects that the lead person should be someone well-versed in modern VU practices and concepts. Conflicts of interest, as potentially would occur if a project modeler or developer were to take the lead in Discovery, should be avoided.

7.2 Example: Discovering the VU Evidence Hierarchy for the Glass-Brine Dissolution Simulation Capability

The Glass-Brine Dissolution Simulation Capability was shown in Figure 2. In Waste Challenge Problem 1, the capability was enhanced by adding a particular glass so that its dissolution in brine could be calculated. In terms of the Discovery process, this capability becomes a landmark simulation capability by adding specific VU objectives (see Figure 9). In talking with modelers on the team, two specific VU objectives were identified as relevant to this capability as it occurs within the THCM ISC hierarchy. The first objective is RP (Research Pedigree), which arises because the capability was imported into the SMP central repository from outside the project. The second objective is VE (Validation to Experiment), because there exists experimental data for glass dissolution. In this example, we establish the VU evidence hierarchy (and VU data) for the VE objective.

7.2.1 The *Representation and Geometric Fidelity* Element

The RCF Element and sub-Element Questionnaires

1. Geometry is irrelevant. Explain. (Placeholder for this explanation).

The RGF sub-Element Questionnaires
Not applicable.

The RGF Simulation Boxes
Not applicable.

7.2.2 The *Math Model Development* Element

The MMD Element and sub-Element Questionnaires

1. What is the physical phenomena? Ans: Chemical Equilibrium (Pointer to)

2. The Global Math Model is Gibb's Free Energy. Characterize this model in mathematical terms. Answer: It is stated as a summation of the energies of each of the species within the solution. As an optimization problem, it is a seven variable continuous optimization problem, with equality constraints. The function which is optimized is known as the Gibbs' Free Energy and consists of a summation of the Gibbs Free Energy models for each of the seven species. The solution to the problem is unique? (Pointer to answer)

(a) The science basis for this model is well-established. (Pointer to the literature)

(b) What are the limits of applicability of this model? (Pointer to answer)

(c) Is GFE appropriate relative to the intended use of this capability. (Pointer to answer).

(d) This model is implemented in the VCS solver within the Cantera Library. (Pointer to pedigree for this solver).

3. There are seven Local Math Models corresponding to the seven species which make up the Glass-Brine solution. They are: SiO_2 (aqueous), SiO_2 (amorphous), Na^+, Cl^-, $OH6-$, H^+, $HSiO_3^-$. How was it decided to use these particular seven species to represent a glass-brine solution for a repository? (Pointer to answer)

4. These local models are non-phenomenological, being curve fits to data. What is the form of the local mathematical models? What are the independent variables and what are the dependent variables? (Pointer to answer).

5. The curve fits are based on calibrations to experimental data, performed by Rimstedt (Pointer to this reference).

6. Why was the Rimstedt model chosen rather than other models? (Pointer to answer)

7. Have the local models been assessed for accuracy? (Pointer to answer)

8. The data used for these calibrations is, by and large, independent of the experimental data that will be used to validate the seven local math models making up the Glass-Brine solution for the present simulation capability. (Pointer to Evidence for this assertion).

9. The coefficients to the curve fits can be found in the Yucca Mountain Thermodynamic Database. (Pointer to a Pedigree for this Database.)

10. The curve fits for the seven species constitute a set of models which must be self-consistent. How is consistency established? (Pointer to answer)

11. Other references. (Pointers to documents containing PIRT and GAP analyses).

The MMD Simulation Boxes There is no plan to perform simulations under MMD because the calibrations were obtained from the literature. How was uncertainty treated in the Rimstedt work? (Pointer to answer).

7.2.3 The *Code Verification* Element

The CV Element and sub-Element Questionnaires

1. Which parts (or Features & Capabilities) of the simulation software will be exercised in making the prediction? (Pointer to answer)

2. What will be the approach in answering the previous question? Answer: So far, we've begun by using a line coverage tool. (Pointer to answer)

3. Will you have a Coverage Table that shows how your code tests cover the Features and Capabilities to be exercised? (Pointer to answer, pointer to coverage table if there is one.) *Note: the code-to-code comparison will use the same inputs as the predictive simulations, so exactly the same code features and capabilities are exercised.*

4. Do the tests in the coverage table establish the correctness of the exercised parts of the software? How? (Pointer to answer)

5. Are there any tests in the Regression Test Suite which can be used to answer the previous questions? If so, is there supporting documentation which explains how the acceptance criterion or fiduciary solution establishes correctness? (Pointer to response) Current unofficial response is that there is no supporting documentation and it is unclear if any of the tests in the Regression Test Suite can provide answers to the previous questions. If there are any such tests, please identify them and the supporting documentation. (Pointer to response)

6. What is your approach to verifying this simulation capability? Answer: We are doing a code-to-code comparison, with the EQ3/6 code.

7. Other references. (Pointers)

The CV Simulation Boxes

Simulation Ensemble: Verification via Cantera to EQ3/6 Comparison.

1. *Pre-Simulation Box.*

   (a) This code was used in the WIPP and Yucca Mountain projects. Since the EQ3/6 code is an auxiliary capability that was not developed in this project, please provide a VU pedigree for it. (Pointer to pedigree)

   (b) Explain why it is valid to compare to EQ3/6 as a means of code verification. (Pointer to explanation)

2. *Simulation Design.*

   (a) To what extent is this an 'apples-to-apples' comparison? (Pointer to answer)
   (b) How many simulations will be in the ensemble and what will be varied across the simulations? (pointer)
   (c) What specific output quantities will be compared? How will they be compared? (pointer)
   (d) (Pointer to simulation design document, if any)

3. *Simulation Ensemble.*

   (a) Will the initial data (input to the software) be checked for correctness? How? (pointer)
   (b) Will the raw output be post-processed? How? Answer: the raw output will be post-processed via a hand-calculation. (pointer)
   (c) How will the raw and processed output be checked for correctness? (pointer)
   (d) (Pointers to the initial data and the output data, and to the post-processor, if any.)
   (e) (Pointer to the computer platform on which these simulations were performed.)

4. *Simulation Analysis.*

   (a) How are the results of the simulations analyzed? Answer: by plotting Solubility vs, pH on a log-log scale. (pointer to explanation)
   (b) How is the comparison performed? (pointer to explanation)
   (c) What was the uncertainty in the input? Is this uncertainty accounted for in the comparison? (pointer)
   (d) (Pointer to analysis document)

5. *Post-Simulation.*

   (a) What is your credible claim concerning the verification of this capability as a result of the code-to-code comparison? (Pointer to answer)
   (b) Has the simulation ensemble been added to the Regression Test Suite? (Pointer to answer)

7.2.4 The *Solution Verification* Element

The SV Element and sub-Element Questionnaires

1. Characterize the numerical algorithm used to solve the Gibbs' Free Energy Equation. Answer: The VCS solver minimizes the total energy of the collection of energies of the seven species by a Picard iteration in which the Hessian matrix has been diagonalized. This algorithm finds stationary points of the objective function. How are the constraints satisfied? (Pointer to answer)
2. How is the iterative algorithm terminated? Answer: by satisfying a tolerance on the residual? How is the residual defined? (Pointer to answer)
3. What types of approximations does the numerical algorithm make in solving the Gibb's Free energy equation? Answer: Numerical roundoff. (Pointer to explanation)
4. What is the accuracy of the simulation output? Answer: Only 3 digits are available in the comparison to experimental data. (Pointer)
5. Was the simulation software verified prior to performing the predictions that are to be compared to experiment? (Pointer to answer)
6. Are there any sensitivity studies that need to be performed in order to characterize the sensitivity to the numerical algorithm parameter input? Answer: Yes, we need to perform sensitivity studies with respect to the tolerance on the residual used to terminate the iterations. (Pointer to answer)
7. Are there error bars around the predictions? If so, how were they determined? (Pointer to answer)
8. Other references. (Pointers)

The SV Simulation Boxes

Simulation Ensemble: Sensitivity to Residual Tolerance.

1. *Pre-Simulation Box.*
   (a) Nothing required here.
2. *Simulation Design.*
   (a) What tolerance values were selected and why? (Pointer to answer)

(b) What are the critical output variables whose sensitivity is to be investigated?

(c) (Pointer to simulation design document, if any)

3. *Simulation Ensemble.*

   (a) Will the initial data (input to the software) be checked for correctness? How? (pointer)

   (b) Will the raw output be post-processed? How? Answer: the raw output will be post-processed via a hand-calculation. (pointer)

   (c) How will the raw and processed output be checked for correctness? (pointer)

   (d) (Pointers to the initial data and the output data, and to the post-processor, if any.)

   (e) (Pointer to the computer platform on which these simulations were performed.)

4. *Simulation Analysis.*

   (a) How are the results of the simulations analyzed? (pointer to answer)

   (b) Did the results of the study result in a a justifiable tolerance for the predictions? (pointer to answer)

   (c) (Pointer to analysis document)

5. *Post-Simulation.*

   (a) What is your credible claim concerning the sensitivity of the critical output variables to the residual tolerance? (Pointer to answer)

   (b) Has this simulation ensemble been added to the Regression Test Suite? (Pointer to answer)

7.2.5 The *Uncertainty Quantification* Element

The UQ Element and sub-Element Questionnaires

1. What are the uncertainties inherent in this VE exercise?

   (a) Missing physics. Answer: see Gap Analysis (Pointer to answer)

(b) The choice of the local math models for the Gibbs' Free Energies of the collection of seven species in the glass-brine solution. Rimstedt vs. Fournier, for example. (This is model form uncertainty) Was this uncertainty accounted for in making the predictions? (Pointer to answer) If so, how was it accounted for? (Pointer to answer) If not, do any additional simulations need to be performed to quantify this uncertainty? (Pointer to answer)

(c) Uncertainty in the model fit coefficients obtained from the database. Was this uncertainty accounted for in making the predictions? (Pointer to answer) If so, how was it accounted for? (Pointer to answer) If not, do any additional simulations need to be performed to quantify this uncertainty? (Pointer to answer)

(d) The numerical approximation uncertainty. Was this uncertainty accounted for in making the predictions? Answer:

   i. The primary uncertainty here is the choice of the value for the residual tolerance. This will be account for in the SV simulations mentioned earlier. (Pointer)

   ii. There is also numerical roundoff and truncation of the answer. (Was three digit accuracy sufficient for the comparison to experiment?) Was this accounted for? If so, how was it accounted for? (Pointer to answer)

   iii. There is also the sensitivity of the solution to the initial guess made to start the iterative solution method. Was this accounted for? If so, how was it accounted for? (Pointer to answer) If not, do any additional simulations need to be performed to quantify these uncertainties? (Pointer to answer)

(e) The uncertainty due to experimental measurements. Was this uncertainty accounted for in doing the validation comparison? If so, how was it accounted for? (Pointer to answer)

(f) The experimental uncertainty as to the exact composition of the glass (were the seven species in this model truly representative of the glass used in the experiment)? Was this uncertainty accounted for in making the predictions? If so, how was it accounted for? (Pointer to answer) If not, do any additional simulations need to be performed to quantify this uncertainty? (Pointer to answer)

2. Other references. (Pointers)

The UQ Simulation Boxes

No additional UQ simulations beyond those performed under MMD and SV have been identified as necessary at this time.

7.2.6 The *Math Model Validation* Element

The MMV Element and sub-Element Questionnaires

1. What experimental data was used? (Pointer to the data) (Pointer to a document describing the experimental setup and results)
2. Has the model accuracy been assessed? (Pointer to answer)
3. Other references. (Pointers)

The MMV Simulation Boxes

Second Simulation Ensemble: Making the Predictions.

1. *Pre-Simulation Box.*
   (a) Nothing required here.
2. *Simulation Design.*
   (a) How well do these simulations represent the experimental setup? (Pointer to answer)
   (b) What residual tolerance value was selected and why? (Pointer to answer)
   (c) What are the critical output variables to be predicted? Are these the same as what was measured in the experiment? (Pointer to answer)
   (d) What validation metrics will be used? (Pointer to answer)
   (e) Has a criterion, based on the chosen validation metric, been established to assess whether the model is valid or not? (Pointer to answer)
   (f) (Pointer to simulation design document, if any)
3. *Simulation Ensemble.*
   (a) Will the initial data (input to the software) be checked for correctness? How? (pointer)
   (b) Will the raw output be post-processed? How? Answer: the raw output will be post-processed via a hand-calculation. (pointer)
   (c) How will the raw and processed output be checked for correctness? (pointer)
   (d) (Pointers to the initial data and the output data, and to the post-processor, if any.)

(e) (Pointer to the computer platform on which these simulations were performed.)

4. *Simulation Analysis.*

   (a) How were the results of the simulations compared to the experimental results? (Pointer to answer)
   (b) Did the comparison use probabilistic methods? (Pointer to answer)
   (c) (Pointer to analysis document)

5. *Post-Simulation.*

   (a) What is your credible claim concerning the validation of this particular glass-brine solution model? (Pointer to answer)
   (b) Has this simulation ensemble been added to the Regression Test Suite? (Pointer to answer)

8 Towards An Implementation of the Discovery-Process Design

In the previous chapter, auxiliary data was identified for a particular VU evidence hierarchy. In this chapter we abstract from this example, as best we can, the kinds of auxiliary data that might be needed in any evidence hierarchy. This list of auxiliary data is preliminary, and can be refined by creating an implementation of our design for the Discovery Process. In the future, the data identified will be used during the Accumulation and Assessment Processes to perform those functions.

As the reader has no doubt noticed by now, the entity-types we have referred to can potentially be thought of, in terms of object-oriented programing concepts, as *classes*. Particularizing one of the entities is equivalent to instantiating a member of the class (i.e., an object). The classes contain place-holders for 'VU-data' (the essential and auxiliary variables) and should also include 'methods' for operating on the data. The Accumulation and Assessment Processes might consist of a collection of methods within or on the classes. Whether or not this is the proper conclusion regarding the entities in the hierarchy is somewhat unclear at this point. It may turn out that some type of database structure is more appropriate. This can best be determined by working on an implementation of this design whilst working on a specific modeling and simulation project.

We list the entity-types and the auxiliary variables that we can foresee might be useful in VU processes that generally would occur after the Discovery process is more or less completed. These additional processes are the Accumulation process (updating data within the hierarchy), and the Assessment process (which operates on static 'snapshots' of the data). Information which can be deduced by traversing the hierarchy is not included in this description.

1. *Landmark Simulation Capability*
   (a) Pointer to an explanation of why this is a landmark simulation capability.
   (b) Pointer to a description of the requirements and/or intended use of the capability.
2. *Simulation Software*
3. *Global Math Model*
   (a) Pointer to the written set of equations in the GMM, along with a reference to its scientific bases and an explanation of it's limitations, particularly with respect to the intended use of the simulation capability.
   (b) Pointer to an explanation of how the global model is solved numerically.
4. *Global Math Model Data*

   (a) Pointer to references for this data.

5. *Local Math Model*

   (a) Pointer to something which indicates whether or not the LMM has been validated to experimental data, and if so, pointers to something describing how it was validated.

6. *Local Math Model Equations*

   (a) Pointer to the written set of equations in the LMM and how it is used within the GMM, along with references characterizing the model and how it was created.

7. *Local Math Model Data*

   (a) Pointer to references for this data

8. *The VU Objective*

   (a) Pointer to the corresponding Element Chain
   (b) Pointer to any VU Requirements pertaining to accuracy of the model or other
   (c) Pointer to a written VU Claim that is made relative to the given objective which is supported by the evidence stored within the hierarchy.

9. *RGF: Representation and Geometric Fidelity Element*

   (a) Boolean to indicate whether this element is relevant. If not relevant, pointer to a written justification as to why it is not relevant.
   (b) Pointer to answers to RGF Element and sub-Element Questionnaires
   (c) Pointer to answers to RGF Simulation Level Questionnaires
   (d) Pointer to a written VU Claim that is made relative to the RGF element, which is supported by the evidence stored within the simulation entities below.

10. *MMD: Math Model Development Element*

   (a) Boolean to indicate whether this element is relevant. If not relevant, pointer to a written justification as to why it is not relevant.
   (b) Pointer to the local math model within the landmark simulation capability
   (c) Pointer to answers to MMD Element and sub-Element Questionnaires
   (d) Pointer to answers to MMD Simulation Level Questionnaires
   (e) Pointer to the data which is to be used in calibration, along with a reference source

(f) Pointer to a written VU Claim that is made relative to the MMD element, which is supported by the evidence stored within the simulation entities below.

(g) Pointer(s) to any PIRT or GAP analyses reports.

11. *CV: Code Verification Element*

    (a) Boolean to indicate whether this element is relevant. If not relevant, pointer to a written justification as to why it is not relevant.

    (b) Pointer to answers to CV Element and sub-Element Questionnaires

    (c) Pointer to answers to CV Simulation Level Questionnaires

    (d) Pointer to a coverage analysis which determines the parts of the code that are traversed using the Simulation Ensemble initial data.

    (e) Pointer to a coverage table showing tests of implementation correctness vs. the parts of the code that are traversed by the simulation ensemble.

    (f) Pointer to written documentation of these tests.

    (g) Variable indicating whether the above tests, when passed, have been added to the regression test suite.

    (h) Pointer to a written VU Claim that is made relative to the CV element, which is supported by the evidence stored within the simulation entities below.

12. *SV: Solution Verification Element*

    (a) Boolean to indicate whether this element is relevant. If not relevant, pointer to a written justification as to why it is not relevant.

    (b) Pointer to answers to SV Element and sub-Element Questionnaires

    (c) Pointer to answers to SV Simulation Level Questionnaires

    (d) Pointer to a written VU Claim that is made relative to the SV element, which is supported by the evidence stored within the simulation entities below.

13. *MMV: Math Model Validation Element*

    (a) Boolean to indicate whether this element is relevant. If not relevant, pointer to a written justification as to why it is not relevant.

    (b) Pointer to the experimental data which is to be used in validating the model, along with references describing the experiment

    (c) Pointer to the specific math model (within the simulation capability hierarchy) that is to be validated

    (d) Pointer to answers to MMV Element and sub-Element Questionnaires

    (e) Pointer to answers to MMV Simulation Level Questionnaires

(f) Pointer to a written VU Claim that is made relative to the MMV element, which is supported by the evidence stored within the simulation entities below.

14. *UQ: Uncertainty Quantification Element*

    (a) Boolean to indicate whether this element is relevant. If not relevant, pointer to a written justification as to why it is not relevant.
    (b) Pointer to answers to UQ Element and sub-Element Questionnaires
    (c) Pointer to answers to UQ Simulation Level Questionnaires
    (d) Pointer to a written VU Claim that is made relative to the UQ element, which is supported by the evidence stored within the simulation entities below.

15. *Pre-Simulation*

16. *Simulation Design*

    (a) Pointer to a document describing the simulation ensemble design.

17. *Simulation Ensemble*

    (a) Pointer to any auxiliary simulation capability used, along with its VU pedigree
    (b) Pointer to any post-processing software used
    (c) Pointer to the computer platform on which the simulations were run

18. *Simulation Analysis*

    (a) Pointer to a document describing the simulation analysis.

19. *Post-Simulation*

    (a) Will the simulation ensemble be repeatable as the software, data, and platforms change with time?

20. *Simulation*

21. *Initial Data*

    (a) Indicator as to whether the initial data was verified, and if so, how.

22. *Raw Output*

    (a) Indicator as to whether the raw output data was verified, and if so, how.

23. *Post-processed Output*

(a) Indicator as to whether the processed output data was verified, and if so, how.

24. *Post-processing Software*

    (a) Pointer to evidence of verification of this software

9 How the Discovery-Process Design Addresses the Requirements

In this short section we re-visit the requirements listed in Section 2 that specify what is needed for a system of VU processes within a large simulation project. We explain how the Discovery process described in this report addresses some of the requirements on the system.

1. *Planning.* The system of processes must assist in the planning of VU work.

Planning is addressed in a major way by the Discovery process. First, it requires the project to define, for each of its ISC's, it's landmark simulation capabilities (those of major VU significance). This is done by the simulation capability development sequence. Second, it requires the project to define for each landmark, the corresponding VU objective. Third, it requires the project to instantiate specific entities within the VU hierarchy so that place-holders for important VU data corresponding to inputs, activities, and results can be managed.

2. *Sequencing.* The system must indicate the order in which the work proceeds and flag any work that is done out of order.

The ISC hierarchy describes an idealized landmark simulation capability development sequence that begins with the project's current state and ends with the planned ultimate end-product or deliverable. The hierarchy provides a de-coupling mechanism between the ordering in which project work is actually performed and a logical ordering upon which a VU narrative can be told. Place-holders within the hierarchy are gradually filled in and, in the project's official repository, the VU data corresponding to the place-holders can be used to reproduce the evidence in a logical order. Each VU objective has a corresponding Element Chain which determines the order in which the VU elements within the particular context are to be addressed. Finally, the five generic VU sub-Elements are strongly ordered. Gaps in the place-holders can be automatically detected so that evidence downstream from the gap is suspect until the gap is filled.

3. *Terminology.* The system must use clear terminology to avoid confusion and provide unambiguous information.

The Discovery process uses well-defined terminology to describe the ISC and VU hierarchies, making clear distinctions (for example) between mathematical models and their implementation in software and between validation of models and validation of software. Although we do not expect nor require that this terminology become universal, a clear and consistent terminology is quite necessary for communication between project members and even stakeholders. Further, we have striven to use terminology which is accepted by at least some part of the VU community. Our terminology is not complete at this point because we have not defined all the processes within the VU system of processes that are needed nor have we drilled down to the lowest level of detail even on the processes which we have defined.

4. *Context.* The system must provide context to define and manage the relationships between components and data within the system.

Context is established primarily through the use of hierarchies which define the relationships between particular objects within the ISC. For example, we do not speak of code verification in the abstract. Rather, after the Discovery process is completed, there will be specific code verification boxes that have been instantiated. Each specific box is part of a specific VU element chain and corresponds to a specific VU objective. In turn, the objective is associated with a specific landmark simulation capability, and thus with a specific set of simulation software (with specific version number). So, one does not ask, has the code been verified? Rather, we ask: has the code been verified with respect to the particular VU objective and simulation capability?

5. *Initiation.* The system must be able to be installed at any point in time within an on-going project and account for any VU work already performed.

The Discovery process can be initiated at any time during the project, although it is clear that the earlier it is initiated, the better, because the ability to impact planning diminishes as initiation of the Discovery process is delayed. As the Discovery process unfolds, it may be found that certain VU activities have already taken place. This is accounted for by the VU objective dealing with inherited (or existing) capabilities. While establishing the pedigree for an inherited capability can be difficult, our process at least accounts for the possibility and thus allows for incorporation of the pedigree evidence into the assessment process. Another likely occurrence in applying the Discovery process is that the simulation capability development sequence (and landmark concept) has not been employed. The Discovery process merely requires that the sequence be defined at initiation; the starting point for the sequence can correspond to whatever stage the project is currently at.

6. *Consonance.* The system must promote appropriate practices within VU, such as those defined in [3] and other VU references.

Consonance is facilitated by the fact that the VU hierarchy is based on the elements (best practices) defined by the PCMM. Additionally, three sets of questionnaires (derived in part from the PCMM) help establish place-holders for important VU data at the Element and Simulation levels in the VU hierarchy. The questionnaires are intended to stimulate thinking about the crucial issues in VU and are fundamental to facilitating consonance in this approach. While the use of appropriate practices cannot be enforced, the assessment process we envision will certainly identify when appropriate practices have not been used.

7. *Concurrent.* The system must be concurrent with other activities that occur as part of the development and use of the modeling and simulation capability.

Ideally, the Discovery process is initiated relatively soon within a modeling and simulation project and is thus concurrent with it. The Discovery process creates place-holders that, as the project proceeds, are filled with concrete data of relevance to VU and its contexts. Thus, additional VU processes such as interim assessment can take place and be concurrent with other project activities.

8. *Practical.* The system must be practical, as demonstrated by its use on at least one concrete example.

The Discovery process is practical in that (1) it was developed by abstracting from the concrete example of Waste, (2) it can be accomplished within a relatively short time (perhaps a month) with respect to the length of the overall project, and (3) it does not make significant demands on project resources (e.g., personnel). If done thoroughly, the Discovery process will likely identify many place-holders and these, during the Accumulation process, may take considerable effort to fill. However, the only process that has input to the quality of the VU data used to fill in the place-holders is the Assessment process, which may drive additional work in filling the place-holders adequately.

9. *Transferable.* The system must be sufficiently general so that it can be applied with ease to more than one concrete example.

The Discovery process has been described abstractly in hope that it can be transferred to other modeling and simulation projects. That is not to say that new issues may not arise as we attempt to apply the process to new projects. However, we are optimistic that any new issues which arise can be accounted for by appropriate extensions of the Discovery process as described here. For example, it may be necessary to add a new VU objective to account for some unusual goal associated with a simulation capability, but it seems unlikely that we will find it necessary to dispose of the concept of a VU objective altogether.

10. *Deliberate.* The system must ensure that the VU work is done deliberately and thoughtfully, as opposed to haphazardly.

The Discovery process sets up place-holders for VU-relevant data and thus, along with the sequencing concept, provides a systematic approach to VU based on the best practices identified by the PCMM.

11. *Multiple Use-Cases.* The system must account for different use-cases that occur within VU.

The concept of a VU objective was created specifically to address this requirement. The seven objectives identified in Section 6.1 essentially correspond to use-cases within VU. The Discovery process identifies these use-cases by establishing the context for the various landmark simulation capabilities within the ISC.

12. *Transparency.* The system must be transparent in that the process and its interim

results are public.

The Discovery process creates place-holders for VU data which will be stored in the Evidence Management System. The plan is for all interested team members to be able to view this data. Further, the data will be used to summarize and publish the ISC hierarchy that was developed with the help of many different members of the team. In this sense, the Discovery process enables transparency of the VU processes.

13. *Records.* The system must identify the appropriate data, evidence, and documentation that can be saved and managed over time.

The use of questionnaire-based place-holders identifies the VU data needed, as shown in the example in Section 7.2.

14. *VU Requirements.* The system must call out the VU requirements that are to be met.

We have devoted little space to how the Discovery process addresses this issue so far. Basically, if an adequate set of Intended Requirements for the modeling and simulation software is given, they would potentially include various VU requirements pertaining to such things as needed accuracy of the model. We anticipate that these kinds of requirements would be associated with the VU objective entity and be used as part of the Assessment process.

15. *Traceability.* The system must provide traceability between VU requirements and the evidence.

With the VU requirements stored within the VU objective entity, trace-ability throughout the hierarchy is ensured.

The remaining system requirements are not, for the most part, addressed by the Discovery process. However, we anticipate that, by establishing the hierarchies, contexts, and VU data pertaining to an given ISC, we are well on our way to defining Accumulation and Assessment processes which can operate on the data and which will address many of the remaining system requirements.

10 Conclusion

A design for a VU Discovery process was proposed in this document and illustrated using the NEAMS Waste THCM Integrated Simulation Capability. The process is based on several abstract hierarchies that define and organize important entities within VU. The Discovery process entails particularizing these abstractions to obtain concrete hierarchies that describe a given modeling and simulation project. Additionally, Discovery entails the identification and creation of place-holders for data within each of the entities within the hierarchy. The place-holders serve both as a planning and a storage mechanism for important VU artifacts and evidence. When completed, the Discovery process basically provides a contract between the VU 'evidence producers' on the one hand, and the VU 'evidence collectors' on the other. That is, after a joint review of the concrete hierarchies and place-holders has been conducted, the evidence-producers have agreed that they will provide the information identified by the place-holders to the evidence-collectors so that the VU story can be told in an organized and comprehensive fashion. The primary purpose of the Discovery process is thus to provide the context and data necessary to perform subsequent VU Accumulation and Assessment processes. We stress that the Discovery process does not provide a recipe for doing VU. The place-holders established in the Discovery process are not intended to take the place of critical thinking about VU nor are they a means of avoiding complexity. VU expertise remains essential to ensuring that the appropriate VU activities are performed and that results are properly interpreted.

Some novel aspects of this design include

1. The use of well-defined hierarchies that provide context and can incorporate levels of fidelity such as sub-continuum, continuum, and performance assessment,

2. The use of place-holders that identify the items needed to tell the VU story,

3. The integration of PCMM concepts such as VU Elements and questionnaires,

4. The use of VU simulation ensembles and simulation box-types within the VU hierarchy so that VU simulations are placed under configuration control,

5. The idea of staying focused on the final product and its associated VU story,

6. The use of the landmark simulation capability development sequence and ISC hierarchy as a means of staying focused on the end-product,

7. The use of a simulation capability hierarchy that provides a clean separation between the math model and the software implementation,

8. The use of VU objectives and Element Chains that are associated with particular simulation capabilities,

9. The use of auxiliary data upon which VU processes can act, and

10. The use of code verification that is performed not on the entire code, but on the capabilities defined by the context provided by the VU objective and hierarchy.

One can, of course, debate whether or not the design we propose is adequate to meet the requirements. The best way to find out will be through implementation and refinement of these ideas. Additionally, in future work, a design for an Assessment process which uses the VU data identified by the place-holders will be defined such that it meets the requirements for Assessment given in Chapter 2.

In closing, context is essential to the application of VU to large simulation projects. In our approach, context is achieved through the use of the VU and ISC hierarchies established by the Discovery process. The importance of context has been recognized in other fields, including non-technical ones. For example, we quote (paraphrasing slightly) comments made by [7] that seem particularly relevant to this report

"... many do not sufficiently appreciate the significance of context. It is easy to misconstrue the context and its importance and hard to know what to do about a changing context full of problems and opportunities. ... contextual information is the primary resource in the sense that without it, we lack even the most elementary tools with which to solve problems. ... mapping context means carrying out descriptive and analytic exercises that reveal the relation between parts and the whole of any problem under consideration. The resulting map is available for all to see, talk about, and use in their deliberations, choices, and implementation. Without a shared, realistic contextual map, all will be operating with incomplete and different maps of what is important and why. ... under-attention to contextual mapping ... severely limits effectiveness, retarding discourse, cooperation, and problem recognition and resolution at important scales. "

Appendix I: VU Element Questionnaires.

E-1.0 Representation and Geometric Fidelity

[E-1.0] *Representation and Geometric Fidelity*. Having to do with the geometry or other representation of the true physical system vs. the geometry of the computational representation of the system. Will over-simplifications in the representation corrupt the simulation conclusions?

To Be Developed.

E-2.0 Math Model Development

[E-2.0] *Math Model Development.* Having to do with the mathematical representation of the physical system, along with its parameters. What is the degree to which the math model is science-based? What is the fidelity of the model?

1. Have the phenomena relevant to this project been identified?

2. Are the math models that describe the phenomena available and implemented in code?

3. Are the math models:

   (i) phenomenological - developed from first principles,

   (ii) non-phenomenological - developed from data

   (iii) other (explain)

4. If the math models are not available, their current state is:

   (i) the theory doesn't exist, models need to be developed from first principles,

   (ii) the basic theory is available but need to be extended to the relevant phenomena,

   (iii) the theory is available but needs to be implemented in code

   (iv) other (explain)

5. For the available math models, how will the parameters be determined:

   (i) using handbooks,

   (ii) using expert opinion,

   (iii) from calibration to available experimental data,

   (iv) the model is non-phenomenological and thus the parameters are non-physical,

   (v) other (explain)

6. Can the math models been assessed for accuracy?

E-3.0 Code Verification

[E-3.0] *Code Verification* Having to do with the correctness and robustness of the computer model. Does the code solve its equations correctly? Is the code robust under the required range of input? Will coding mistakes or algorithmic deficiencies corrupt the simulation results?

To Be Developed.

E-4.0 Solution Verification

[E-4.0] *Solution Verification*. Having to do with the integrity of the input and output of the computer model when the latter is used in validation or to make a prediction. Was the input to the computer model correct? Have the discretization and round-off errors been characterized?

1. Will the code(s) be verified (according to E-4.0) before they are used to generate the solutions?

2. Does the code or codes produce the numerical solution by solving a differential equation?

E-5.0 Math Model Validation

[E-5.0] *Validation.* Having to do with comparison of computer model output and experimental results. To what degree do these match?

1. Does the model accuracy need to be assessed?

2. Is there data available to conduct an assessment?

3. How relevant is the data used for validation relative to the intended application?

(i) Very relevant (the data used for validation is very close to the application space),

(ii) Mildly relevant (close to the application but some extrapolation will be needed),

(iii) Not relevant (validation data is far away from the application space

4. What are the quantities of interest that you will be comparing from test data and simulation data?

5. What is the source of the data?

(i) Relevant experimental data,

(ii) Historical data,

(iii) Expert opinion,

(iv) A combination of all/some of the above (which ones)

(v) Other (explain)

6. Do you have uncertainties on the test data? If not, how do you plan to compute/estimate uncertainties on the test data?

7. Have validation metrics (i.e. the metrics used to compare and assess the model "validity) been defined for this problem?

(i) are they probabilistic in nature,

(ii) are they ad-hoc

8. Has a criteria based on the defined validation metric, been established to assess whether the model is valid or not?

9. How do you plan to compare test data (with its uncertainties) to simulation data (with its uncertainties)

(i) by plotting simulation vs. experimental data on the same plot (viewgraph norm),

(ii) by comparing amount of error between experimental and simulation data,

(iii) by doing probabilistic comparisons (and/or test of hypothesis)

(iv) other (list)

10. Assess the degree of interpolation or extrapolation

E-6.0 Uncertainty Quantification

[E-6.0] *Uncertainty Quantification*. Having to do with sensitivity analysis, uncertainty propagation, and margins. What is the impact of variability's and uncertainties on system performance and margins?

1. Have you done a sensitivity analysis to help identify the most relevant parameters in the model?

2. Have you quantified sources of uncertainties, types and characterization of their probability model? If so, check those which apply:

(i) Material properties,

(ii) Boundary and initial conditions

(iii) Empirical inputs

(iv) Model form uncertainty

(v) Missing physics

(vi) Solution error

(vii) Geometric and modeling assumptions,

(viii) other (list)

3. How will you obtain data to quantify model uncertainties?

(i) Develop and perform experiments

(ii) Subject matter expert opinion

(iii) Legacy data

(iv) Handbooks/literature search

(v) Other (explain)

4. Have you quantified experimental uncertainties, types and characterization of their probability model. Some examples are:

(i) Measurement errors measurement techniques and post-processing

(ii) Unit-to-unit variability

(iii) Test-to-test variability

(iv) Boundary conditions and inputs

(v) Experimental biases

(vi) Test environment vs. actual environment

5. Have sources of epistemic uncertainty being identified? Will they be propagated? How?

6. Has the use of surrogate models for efficient uncertainty propagation being considered?

7. What are the computational resources necessary to propagate uncertainty? Are they sufficient?

8. What software dependencies are necessary to enable uncertainty quantification and sensitivity analysis?

9. Will the UQ process be peered review and documented?

Appendix II: VU sub-Element Questionnaires. For each relevant sub-Element, answer the following questionnaires.

E-2.1 *Identify and rank the applicable physical phenomena.*

Having to do with determining which physical phenomena should be modeled and their relative importance

5-1. When this sub-element is completed, what will you be able to credibly claim ?

(Suggested Claim): The phenomena relevant to the target application has been identified and ranked in order of importance. In addition, the available models that represent this phenomena has been identified and modeling gaps have been identified and prioritized.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- the list of relevant phenomena,
- the list of relevant phenomena, along with an explanation,
- a document containing the list and an explanation,
- identification of FEPs, PIRTs, or other general lists of phenomena which were used to identify the relevant phenomena
- a ranked list, with explanation of how the particular rankings were determined
- other

5-3. What will be the Approach?

a) Master Phenomena List:

b) Method of Down-selecting:

- expert opinion,
- consultation of literature,
- other

c) Method of Ranking:

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Date:

b) Person filling out this form:

c) name of person(s) responsible for creating the evidence:

d) If any of the evidence identified above already exists, please upload it now. Give the list of phenomena now and specify on what page of the document it can be found.

E-2.2 *Determine the Math Models from the Phenomena.*

Having to do with acquiring, establishing, or constructing a mathematical model of the phenomena. What math models currently exist for the phenomena of interest? What is the pedigree of the model(s)? Are the models adequate? Do any of the models need to be changed or improved? Are any new models needed?

5-1. When this sub-element is completed, what will you be able to credibly claim ?

(Suggested Claim): Given the phenomena identified in E-2.1 and the priority ranking obtained from expert elicitation, a mathematical mode(s) has been identified and/or developed to address the phenomena of interest. The mathematics have been implemented into a code and integrated into an analysis software.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- Report documenting math model development and code development
- Documentations for existing code detailing math model form, assumptions and limitations

5-3. What will be the Approach?

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Date:

b) Person filling out this form:

c) name of person(s) responsible for creating the evidence:

E-2.3 *Parameter Estimation and Calibration.*

Having to do with the determination of non-stochastic physical parameters in the model.

5-1. When this sub-element is completed, what will you be able to credibly claim ?

(Suggested Claim): The estimated parameters, along with the math model developed/identified in E-2.2, can be used with confidence over an identified range of inputs. Confidence is based on a comparison between 'goodness of fit' of the data to the model with an acceptance criterion which is based on the intended use of the model.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- a pedigree for the fitted data,
- 'goodness of fit' measurements
- rationale for choice of acceptance criterion
- an explanation for the stated parameter range under which the model is valid,
- a discussion of the assumptions and limitations in the work,
- other

5-3. What will be the Approach?

a) Type of data to be fitted (check one or more):

- none,
- experimental,
- code-generated,
- tables,
- scientific,
- other

b) Type of math model used:

- curve fit,
- science-based,
- first principles,
- other

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) What parameters will be estimated?

b) What model(s) are you using to estimate the parameters from?

c) Date:

d) Person filling out this form:

e) name of person(s) responsible for creating the evidence:

E-3.1 *Features and Capabilities List.*

Having to do with the identification of code functionalities that are relevant to the intended use or functional requirements for the code or module *after* the current M&S effort is completed. Examples of code 'features' are solver options, mesh element-type options, discretization options, limiter and other numerical options. Example of code 'capabilities' are boundary condition types, material model options, closure relations, and source term models. Includes combinations of features and capabilities.

5-1. When this sub-element is completed, what will you be able to credibly claim?

(Suggested Claim): A list of all code functionalities was complied. Each functionality was assessed in terms of whether or not it is related to a functional requirement of the code. Combinations of functionalities related to functional requirements were also identified.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- a reference giving the functional requirements of the code,
- a description or list of the functional requirements for the code,
- a descriptive list of all the code functionalities,
- a table or description showing relationship between specific features and capabilities to specific functional requirements
- a discussion of the assumptions and limitations of the list
- other

5-3. What will be the Approach?

a) How will (or were) the functional requirements be determined?

- not determined,
- from the customer,
- from the modelers/analysts,
- from a PIRTS or FEPS analysis
- other

b) How will (or were) the code functionalities be determined?

- from an existing list,
- from the test suite,
- from the users manual,
- from the theory manual,
- from the source code,
- expert opinion,
- other

c) What will be the level of detail in the list of code functionalities?

1. coarse grained,
2. medium grained,
3. fine grained,
4. varying

d) What is or will be the relationship between code functionalities and functional requirements?

1. a one-to-one map
2. more functionalities than requirements
3. more requirements than functionalities

e) How will the relationship between code functionalities and functional requirements be determined?

1. expert opinion
2. other

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Should (or will) irrelevant code features and capabilities be dis-abled?

b) Date this form was filled out:

c) Person filling out this form:

d) name of person(s) responsible for creating the evidence:

E-3.2 *Gold Copy Tests.*

Having to do with acquiring, creating, passing, and documenting tests of the relevant code functionalities.

5-1. When this sub-element is completed, what will you be able to credibly claim?

(Suggested Claim): Every relevant code functionality identified in E-4.1 has one or more associated gold copy tests that have been passed and documented.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

1. a document describing each test,
2. an independent review of the test documentation,
3. other (*fill in blank*).

5-3. What will be the approach?

a) Each test will be documented as follows: (check one or more)

1. an explanation of which features and capabilities the test covers,
2. the test is classified according to purpose (capability demonstration, robustness, correctness, other),
3. the test is classified according to type (unit, functional, other.)
4. an acceptance criterion,
5. a description of what physical problem is solved,
6. a description of what mathematical problem is solved,
7. the precise input is saved,
8. the precise output is saved,
9. the person(s) or team which created the test is identified

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Date this form was filled out:

b) Person filling out this form:

c) name of person(s) responsible for creating the evidence:

E-3.4 *Code Coverage Status Tables.*

Having to do with identifying the degree to which various features and capabilities have been tested. This is not a test plan, but a status table.

5-1. When this sub-element is completed, what will you be able to credibly claim?

(Suggested Claim): Every code feature and capability from E-3.1 will be covered by one or more test for which there is a documented gold copy, is classified according to test type, and which resides in the regression test suite.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- sub-Element 3.1 has been completed,
- sub-Element 3.2 has been completed,
- sub-Element 3.3 has been completed,
- a coverage status table will be produced
- a coverage status table for this code can be found in the EVIMS
- the coverage status table is periodically updated
- other

5-3. What will be the Approach?

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Date this form was filled out:

b) Person filling out this form:

c) name of person(s) responsible for creating the evidence:

E-4.1 *I/O Verification.*

t Having to do with establishing that the code inputs and outputs were valid.

5-1. When this sub-element is completed, what will you be able to credibly claim ?

(Suggested Claim): The code I/O was verified by:

(a) the person(s) who created it, or
(b) a team peer, or
(c) someone external to the team, or
(d) independently reproducing it.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- expert opinion,
- a written discussion on how the I/O was verified,
- other

5-3. What will be the Approach?

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Date:

b) Person filling out this form:

c) name of person(s) responsible for creating the evidence:

E-4.2 *Numerical Model Sensitivity Studies.*

Having to do with sensitivity of the computer model output to non-physical input parameters (e.g., solver tolerances, convergence criteria, artificial damping parameters).

5-1. When this sub-element is completed, what will you be able to credibly claim ?

(Suggested Claim): We understand the sensitivity of our critical system response quantities to *all* the non-physical input parameters. Values for these parameters were selected based on the sensitivity study.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- plots of SRQ's vs. each of the non-physical input parameters
- a written rationale for why the final values of these parameters were chosen
- uncertainty bars centered around the final values
- other

5-3. What will be the Approach?

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Date:

b) Person filling out this form:

c) name of person(s) responsible for creating the evidence:

d) If any of the evidence identified above already exists, please upload it now.

E-4.3 *Mesh Refinement Studies.*

Having to do with the sensitivity of the computer model output to the discretization.

5-1. When this sub-element is completed, what will you be able to credibly claim ?

(Suggested Claim): We understand the sensitivity of our critical system response quantities to both the mesh resolution and the time-step size. Values for the mesh and time-step size were selected based on the sensitivity study.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- Evidence showing that the numerical solution is (or is not) in the asymptotic range,
- Evidence showing that the mesh quality is sufficient,
- Plots of critical system response quantities vs. mesh and time-step size,
- Other

5-3. What will be the Approach?

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Date:

b) Person filling out this form:

c) name of person(s) responsible for creating the evidence:

d) If any of the evidence identified above already exists, please upload it now.

E-4.4 *Error Estimation.*

Having to do with estimating the discretization error of the numerical solution.

5-1. When this sub-element is completed, what will you be able to credibly claim ?

(Suggested Claim): The discretization error has been characterized for each of the computed values of the critical system response quantities that will be used for Validation or UQ.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- Error bars around the critical system response quantities,
- Error estimates applied to numerical solution,
- Expert Opinion,
- Other

5-3. What will be the Approach?

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Date:

b) Person filling out this form:

c) name of person(s) responsible for creating the evidence:

d) If any of the evidence identified above already exists, please upload it now.

E-5.1 *Apply a Validation Hierarchy.*

Having to do with the concept of building large system level models in a hierarchical or building block approach. The main idea of a validation hierarchy is that confidence in the models is established by validating models at the component level (smaller single block(s)), then validating at the sub-system level (i.e. groups of components) and finally at the system level (if possible).

5-1. When this sub-element is completed, what will you be able to credibly claim ?

(Suggested Claim): Based on the application of interest, a validation hierarchy was identified and model were assess following the hierarchy. System level models were constructed in a hierarchical manner and thus have been assessed in this manner.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- Report documenting hierarchical system level model building approach
- Report documenting individual model validation assessments done at each level of the hierarchy

5-3. What will be the Approach?

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Date:

b) Person filling out this form:

c) name of person(s) responsible for creating the evidence:

d) If any of the evidence identified above already exists, please upload it now.

Form E-5.2 *Assess the relevance of the underlying database used for model validation.*

Having to do with establishing how relevant is the data used to assess the model predictive capability relative to the end use of the model. It is also an attempt to define the limits of interpolation and extrapolation for the model.

5-1. When this sub-element is completed, what will you be able to credibly claim ?

(Suggested Claim): A database exists that provides relevant data to assess the model. The data has been assessed and confirmed that its pedigree can be established and the space covered by the data is appropriate for the application space that the model will be used in. Alternatively, the data is not inclusive of the space of application of the model but the data is sufficient to establish a boundary between interpolation and extrapolation.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- A database suitable for use in model validation assessment
- Documentation establishing the pedigree of the database
- Documentation establishing uncertainties in the data collected
- Documentation establishing the assessment of the database, the metrics used and criteria by which this database was deemed relevant.

5-3. What will be the Approach?

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Date:

b) Person filling out this form:

c) name of person(s) responsible for creating the evidence:

d) If any of the evidence identified above already exists, please upload it now.

E-5.3 *Define model validation metrics and criteria for model adequacy.*

Having to do with establishing the metrics and criteria (based on these metrics) that will be used to assess the model. The sub-element refers to the model being assess for adequacy for the intended application and thus the metrics and criteria should be influenced by this end use.

5-1. When this sub-element is completed, what will you be able to credibly claim ?

(Suggested Claim): Validation metrics and appropriate criteria for model assessment have been developed and are documented. The metrics and criteria are established and set relative to the application environment of the model.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- Document detailing development of validation metrics and assessment criteria
- Document detailing target application of the model

5-3. What will be the Approach?

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Date:

b) Person filling out this form:

c) name of person(s) responsible for creating the evidence:

d) If any of the evidence identified above already exists, please upload it now.

E-6.1 *Establish framework for dealing with uncertainties.*

Having to do with creating a framework to incorporate and propagate sources of aleatoric and epistemic uncertainty. This framework takes into account the hierarchical nature of the modeling effort as well as the multi-levels of fidelity both in spatial and temporal resolution.

5-1. When this sub-element is completed, what will you be able to credibly claim ?

(Suggested Claim): A framework has been developed that incorporates both aleatoric and epistemic sources of uncertainty and provides a clear distinction of the effect of each one on the quantity of interest. Further, the framework is capable of propagating uncertainty among different level of fidelity both in spatial and temporal resolution.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- A document describing the mathematical formulation of the framework used
- An implementation of the framework and the documentation of the tool created to incorporate the framework (e.g. DAKOTA and its user manuals)

5-3. What will be the Approach?

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Date:

b) Person filling out this form:

c) name of person(s) responsible for creating the evidence:

d) If any of the evidence identified above already exists, please upload it now.

E-6.2 *Perform Sensitivity Analysis.*

Having to do with performing sensitivity analysis to establish which parameters are most sensitivity to the response of interest. In addition, this sub-element will yield which parameters will contribute more to the uncertainty in the response of interest.

5-1. When this sub-element is completed, what will you be able to credibly claim ?

(Suggested Claim): A sensitivity analysis was completed and the parameters which contributes the most to the variability of the response quantity of interest were identified. .

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- A document describing the approach to perform the sensitivity analysis
- The tool and its documentation used to perform the analysis

5-3. What will be the Approach?

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Date:

b) Person filling out this form:

c) name of person(s) responsible for creating the evidence:

d) If any of the evidence identified above already exists, please upload it now.

E-6.3 *Quantify numerical propagation errors.*

Having to do with quantifying error due to modeling a physical phenomena. These errors could arise from mesh discretization, multiple models to represent the same phenomena, solution convergence errors, etc.

5-1. When this sub-element is completed, what will you be able to credibly claim ?

(Suggested Claim): Sources of numerical propagation errors were identified and quantified. These errors come from sources such as mesh discretization, the presence of multiple models to capture the same phenomena, solution convergence errors and other sources that are relevant to the intended application.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- A document describing the sources of error and the way they were quantified

5-3. What will be the Approach?

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Date:

b) Person filling out this form:

c) name of person(s) responsible for creating the evidence:

d) If any of the evidence identified above already exists, please upload it now.

E-6.4 *Quantify experimental/data uncertainties.*

Having to do with quantifying uncertainties in the experimental data and any data obtained to support model creation and/or validation. These uncertainties could include inherent variability in the system of interest, variability due to test to test conditions, boundary conditions, environmental conditions, etc.

5-1. When this sub-element is completed, what will you be able to credibly claim ?

(Suggested Claim): The data that will be used to create the model and for model validation has been identified and sources of uncertainties related to this data have been identified and quantified. Sources of uncertainty include measurement error, material variation information and uncertainties due to boundary conditions. Sources of uncertainties were also identified and quantified for data obtained from literature reviews and expert opinion.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- Document describing the data, its pedigree, sources and quantification of uncertainty
- The repository where the data is stored and documentation describing the data format and underlying source

5-3. What will be the Approach?

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Date:

b) Person filling out this form:

c) name of person(s) responsible for creating the evidence:

d) If any of the evidence identified above already exists, please upload it now.

E-6.5 *Aggregate evidence for uncertainties.*

Having to do with combining the identified sources of uncertainty and using this knowledge to obtain a best prediction value plus uncertainty.

5-1. When this sub-element is completed, what will you be able to credibly claim ?

(Suggested Claim): The various sources of uncertainty that were identified and quantified in Sub-element 6.1 and 6.2 have been combined in a reasonable way to obtain an estimate of the total uncertainty. This total uncertainty will be used to make an estimate of best prediction plus total uncertainty.

5-2. What evidence will be produced to support this claim?

Suggested evidence (check one or more of the following):

- Documentation of the sources and method for quantification of uncertainty
- Documentation of the methodology to combine the sources of uncertainty
- Documentation of the tool(s) used to estimate the total uncertainty

5-3. What will be the Approach?

5-4. Temporal Aspects of this Work:

a) What tasks, VU Elements, or VU sub-Elements should be completed before this sub-Element can be addressed?

b) What tasks, VU Elements, or VU sub-Elements cannot be completed before this one is performed?

5-5. Additional Information:

a) Date:

b) Person filling out this form:

c) name of person(s) responsible for creating the evidence:

d) If any of the evidence identified above already exists, please upload it now.

Appendix III: Simulation Level Questionnaires

The RGF Element (E-1.0) Reserved for future work.

The MMD Element (E-2.0)

S-2.1 *MMD Practices Required Prior to Calibration.*

- *Identify and rank the applicable physical phenomena.*
- *Determine Math Model(s) from the Phenomena.*

S-2.2 *Design the Calibration Runs.*

- *What experimental data will be included in the calibration?*
- *What math model will the data be fit to?*
- *What are the key physical parameters?*
- *How will the comparisons be made?*
- *How will uncertainty in the experimental data be incorporated?*
- *How will numerical model uncertainty be incorporated?*

S-2.3 *Apply Good Simulation Practices.* How will the input and output be verified?

S-2.4 *Analyze and Explain the Calibration Results.*

S-2.5 *MMD Practices Required Post-Simulation.* None.

The CV Element (E-3.0)

S-4.1 *Create the Features and Capabilities List.* Identify the code functionalities that are relevant to the context within which the CV element occurs.

S-4.2 *Design the Simulation Ensemble (Test Suite).*

- *How will the features and capabilities be tested?*
- *What is the purpose of each test?*
- *What is the acceptance criteria for each test?*
- *Can any of the existing tests in the Regression Test suite be used?*

S-4.3 *Apply Good Simulation Practices.* How will the input and output be verified? (Or, how was it verified?)

S-4.4 *Analyze and Explain the Simulation Results.* Document each simulation in terms of input, output, comparison of results to acceptance criterion. Create a code coverage status table.

S-4.5 *CV Practices Required Post-Simulation.* Ensure that the gold copy tests become part of a regression testing procedure.

The SV Element (E-4.0)

S-4.1 *SV Practices Required Prior to Making the Prediction.* None.

S-4.2 *Design the End-Use Predictive Simulation Ensemble.*

- *What are the key physical input parameters?*
- *What are the key physical output parameters?*
- *How will sensitivity of the key output parameters to the numerical model parameters be incorporated?*
- *How will solution convergence be established?*
- *How will discretization error be incorporated and in what output variables?*
- *Will error bars be included?*

S-4.3 *Apply Good Simulation Practices.* How will the input and output be verified? (Or, how was it verified?)

S-4.4 *Analyze and Explain the Simulation Results.* Were the solutions converged? Is the discretization error sufficiently characterized?

S-4.5 *SV Practices Required Post-Simulation.* None.

The MMV Element (E-5.0)

S-5.1 *Assess the relevance of the underlying database.* Having to do with establishing how relevant or valid is the data used to assess the model predictive capability.

S-5.2 *Design the Simulation Ensemble.* This was already accomplished in the SV element.

S-5.3 *Analyze and Explain the Comparison Between the Prediction and the Experimental Data.* Define model validation metrics and criteria for model adequacy. Having to do with establishing the metrics and criteria (based on these metrics) that will be used to assess the model. Document the comparison and explain.

The UQ Element (E-6.0) Reserved for future work.

References

[1] W. Oberkampf and C. Roy, *Verification and Validation in Scientific Computing*, Cambridge University Press, 2010.

[2] P.J. Roache, *Fundamentals of Verification and Validation*, Hermosa Pub., Socorro NM, 2009.

[3] R. Nelson, C. Unal, J. Stewart, B. Williams, *Using Error and Uncertainty Quantification to Verify and Validate Modeling and Simulation*, Los Alamos National Laboratory, LA-UR-10-06125, 2010.

[4] H.C. Edwards et. al. *Nuclear Energy Advanced Modeling and Simulation Waste Integrated Performance and Safety Codes - Verification and Validation Plan*, SAND2011-0084, Sandia National Laboratories, 2011.

[5] W. L. Oberkampf, M. Pilch, and T.G. Trucano, *Predictive Capability Maturity Model for Computational Modeling and Simulation*, Sandia National Laboratories, SAND2007-5948, 2007.

[6] M. Pilch, T. Trucano, and K. Alvin, *Assessing Predictive Capability*, manuscript, Sandia National Laboratories, 2011.

[7] S.G. Clark, *Ensuring Greater Yellowstone's Future*, Yale Univ. Press, New Haven, 2008.

DISTRIBUTION:

1	MS 0828	Ken Alvin, 1544
1	MS 0372	Jose Arguello, 1525
1	MS 0828	Ryan Bond, 1541
1	MS 1138	Julie Bouchard, 6923
1	MS 0897	Kevin Copps, 1544
1	MS 0828	Kevin Dowding, 1544
1	MS 1318	Carter Edwards, 1444
1	MS 1370	Geoffrey Freeze, 6224
1	MS 0779	Carlos Jove-Colon, 6222
1	MS 1318	Rob Hoekstra, 1426
1	MS 1318	Patrick Knupp, 1441
1	MS 1322	Vincent Mousseau, 1444
1	MS 0316	Curtis Ober, 1442
1	MS 0828	Martin Pilch, 1514
1	MS 0378	William Rider, 1443
1	MS 0378	Allen Robinson, 1443
1	MS 1322	Peter Schultz, 1425
1	MS 1318	James Stewart, 1441
1	MS 1321	Randy Summers, 1444
1	MS 1318	Laura Swiler, 1441
1	MS 0370	Tim Trucano, 1441
1	MS 0828	Angel Urbina, 1544
1	MS 1322	Dena Vigil, 1444
1	MS 0779	Yigeng Wang, 6222
1	MS 0378	Greg Weirs, 1441
1	MS 0899	Technical Library, 9536, electronic copy

Sandia National Laboratories