Verification, Validation, and Accreditation (VV&A)

VV&A applies to the verifying and validating models and simulations (M&S) used in the conduct of a JT&E. Whenever the use of M&S is anticipated in a JT&E, the PTP should include a section that addresses VV&A plans. The rigor associated to the VV&A of M&S that will be used to support JT&E programs should be based on a number of factors. If a simulation has been used in other related applications and has undergone a VV&A process, it may be more appropriate to accept the previous accreditation if it has been methodically documented and the JT&E Director is convinced that the JT&E application is sufficiently similar to the application for which the M&S was accredited. In the event that a full VV&A must be performed, the JT&E Director must determine the scope of the VV&A process in consonance with any budgetary constraints. New simulations that have been well documented may not require extensive verification of the simulation code. The JT&E Director may be satisfied with a previous verification process and may desire to conduct a validation of the M&S. It is recommended that before any decision is made that may affect a budgetary decision, especially for the APA, the JT&E Director and his staff should become familiar with the contents of the VV&A Recommended Practices Guide.

Department of Defense

Verification, Validation and Accreditation (VV&A) Recommended Practices Guide

November 1996

Office of the Director
of Defense Research and Engineering
Defense Modeling and Simulation Office

MEMORANDUM FOR: GENERAL DISTRIBUTION

SUBJECT: Verification, Validation, and Accreditation (VV&A)
Recommended Practices Guide

I commend to you the attached guide which provides background and information on principles, processes, and techniques which are recommended for use in DoD VV&A efforts which support program initiatives in the analysis, acquisition, and training communities.

These guidelines reflect a year-long study of Service directives and VV&A techniques from government, industry and academia. An integrated team of DoD-recognized VV&A experts authored the Guide and obtained informal coordination throughout its development from contributors across DoD.

The guide will continue development during Fiscal Year 1997 to include more detailed guidance for VV&A efforts performed to support modeling and simulation in the three functional areas of analysis, acquisition, and training.

Please call Mrs. Priscilla Glasow, DMSO Technical Support Staff, at 703-824-3412, or complete the evaluation form at the back of this document if you have any questions or suggestions for improvement.

James W. Hollenbach
Captain, U.S. Navy
Director, Defense Modeling and Simulation Office

Contents

List of Figures

List of Tables

Acronym List

AAR	after action review
ALSP	Aggregate-Level Simulation Protocol
ANSI/IEEE	American National Standards Institute/Institute of Electrical and Electronics Engineers
AR	Army Regulation
CDR	Critical Design Reviews
c.i.	confidence interval
CINCs	Commanders in Chief
C/M	Configuration Management
CMMS	Conceptual Models of the Mission Space
CPU	central processing unit
DARPA	Defense Advance Research Projects Agency
DIS	Distributed Interactive Simulation
DLA	Defense Logistics Agency
DMSO	Defense Modeling and Simulation Office
DoD	Department of Defense
DoDD	Department of Defense Directive
DoDDI	Department of Defense Directives and Instructions
DoDI	Department of Defense Instruction
ECM	electronic countermeasures
EMF	exercise management and feedback
FOMs	Federation Object Models
FRED	Federation Required Execution Details
HLA	High-Level Architecture
IAC	Information Analysis Center
IPR	In-Process Reviews
IST	Institute for Simulation Training
IV&V	Independent Verification and Validation
JTC	Joint Training Confederation
M&S	Modeling and Simulation
MOEs	Measures of Effectiveness
MOMs	Measures of Merit (MOMs encompass MOEs, MOOs, and MOPs)

MOOs	Measures of Outcome
MOPs	Measures of Performance
MORS	Military Operations Research Society
m.r.a.	model range of accuracy
MSRR	Modeling and Simulation Resource Repository
PDR	Preliminary Design Reviews
PDU	Protocol Data Units
ROI	return on investment
RTI	Run Time Infrastructure
s.c.i.	simultaneous confidence intervals
SEI	Software Engineering Institute
SIMNET	Simulator Networking
SMART	Susceptibility Model Assessment with Range Test
SME	subject matter experts
SOMs	Simulation Object Models
STRICOM	Simulation, Training, and Instrumentation Command
T&E	test and evaluation
UJTL	Uniform Joint Task List
V&V	Verification and Validation
VV&A	Verification, Validation, and Accreditation
VV&C	Verification, Validation, and Certification

Meet the Authors

In early 1995, the Defense Modeling and Simulation Office (DMSO) formed a VV&A Technical Support Team to develop and write a guide of recommended VV&A practices for the Department of Defense. The VV&A Technical Support Team was initially formulated to include representation of early DoD experience in federating models and simulations, such as legacy models, Distributed Interactive Simulation (DIS), and new model developments. Subsequent additions to team membership ensured informal representation of the military services, academia, and industry. The Guide as written today successfully incorporates existing directives and guidelines available from all of these sources.

This section provides the reader with brief descriptions of the authors and their credentials in the VV&A community.

Mrs. Priscilla A. Glasow, the Guide's Editor, originated the concept of a Technical Support Team to author a guidelines document to meet the requirements of the DoD Modeling and Simulation Master Plan. She successfully organized and led a team of proven VV&A experts from all defense communities, providing clear management as DMSO's direct representative and herself authoring significant portions of key chapters. Mrs. Glasow is a Senior Systems Analyst with Science Applications International Corporation and serves on DMSO's Technical Support Staff.

Dr. Paul Muessig of the Susceptibility Model Assessment and Range Test (SMART) Project Office in China Lake, California, is one of the original members of the team since its inception and has provided clear leadership in the development of the Guide. He has served as both a representative of the acquisition community and the U.S. Navy. The SMART process was used as a baseline study from which the generic VV&A process and the methodology described herein were developed. The primary author of Chapter 1, Dr. Muessig also contributed significantly to the other chapters and to the discussions which guided the development of the Guide in its conceptual stage.

Similarly, the Guide reflects the invaluable wisdom and guidance of Mr. James Sikora, Senior Vice President and General Manager of BDM Management Services Company in Albuquerque, New Mexico. Mr. Sikora is well known for his co-chairmanship of the Simulation Validation series of workshops under the Military Operations Research Society (MORS). It was the initial workshop under this series that first formulated the definitions for VV&A that were later adopted by DoD and are in use today. Mr. Sikora's representation of the analytic community and his long-term close association with the U.S. Air Force's Operational Test and Evaluation Command (AFOTEC) ensured that the Guide met the requirements of these communities as well. Mr. Sikora authored Chapters 5 and 6 and provided sage overall perspective to the team in its development of the other chapters.

A third key player in the development of the Guide was Ms. Simone Youngblood of Illgen Simulation Technologies, Inc. Ms. Youngblood has served as the Chair of the VV&A sub-working group in the DIS Workshop for many years and is widely recognized as the leading author of the DIS Nine-Step VV&A Process. Serving as both a DIS community representative and that of industry, Ms. Youngblood authored Chapter 3.

Dr. Osman Balci, Associate Professor of Computer Science at Virginia Polytechnic University, is a well-known and widely respected author of numerous treatises on software verification, validation, and testing. As the team's academic representative, he provided a significantly new and different perspective and ensured much re-thinking of the team's objectives and solutions. Dr. Balci shared his extensive background in his writing of Chapters 2 and 4.

Ms. Susan Solick of the U.S. Army's TRADOC Analysis Command located at Fort Leavenworth, Kansas, and Dr. Ernie Page of MITRE Corporation were key contributors to the development of Chapter 3 and the descriptions of the DIS and Aggregate Level Simulation Protocol (ALSP) processes. Ms. Solick has worked with the DIS community for the past 3 years and served as the team's U.S. Army representative. Dr. Page is the a leading member of the VV&A Technical Staff for the ALSP Joint Training Confederation.

The contributions of earlier members of the Technical Support Team and leaders in the VV&A community are also noted. Ms. Pam Blechinger served on the Technical Support Team in its first year and was an active participant in the discussions which resulted in the initial draft of the Guide. Mr. Robert Lewis of Quality Research, Inc. of Huntsville, Alabama was a leading source of VV&A theory and is the author of a tailoring and costing study which has served as the basis for that work in this Guide. Mr. Chuck Winget of Illgen Simulation Technologies, Inc. was also a frequent contributor and provided a valuable assessment of Mr. Lewis' costing model.

Making the Best Use of This Document

The Department of Defense (DoD) Verification, Validation, and Accreditation (VV&A) *Recommended Practices Guide* is written for use by all developers and users of Modeling and Simulation (M&S) in DoD. This general audience is divided into three loosely defined "groups"—decision makers, program managers, and technical staff—a distinction that merely serves to define different levels of involvement in the VV&A process.

The Chapters in Brief

All readers will be interested in the Chapter 1 overview, particularly those sections dealing with the benefits of doing VV&A and tailoring it to contain costs.

Chapter 2 discusses basic principles of VV&A and provides amplification of the major points contained in Chapter 1.

Chapter 3 introduces a generic VV&A process and discusses its relationship to various types of M&S applications, including the High-Level Architecture (HLA). This chapter will be of particular interest to program managers who must integrate VV&A into their overall programs.

Chapter 4 is the technical meat of the guide, offering technical staff a host of fundamentals and techniques for performing VV&A and helping readers determine which techniques are most useful for specific types of M&S application. This section will be greatly expanded as programs mature and case studies become available.

Chapter 5 discusses the accreditation process and the work that must be done to reach a sound decision about the suitability of M&S for particular applications. It is an excellent chapter to guide the decision maker on how to plan for and implement the accreditation process and on how to integrate V&V into the decision.

Finally, Chapter 6 introduces common reporting formats for the reports that should document any VV&A effort. Although each Branch of Service may prescribe the reports it requires, this chapter provides formats that meet the common needs of all Services and thus are particularly useful when M&S is applied to a Joint requirement.

Recommended Reading for Specific Needs

Chapters 1 and 5 are recommended for decision makers who need a quick overview of VV&A and information on the accreditation process.

Program managers are referred to Chapters 1, 2, 3, and 5. Again, a quick overview of what VV&A is all about is a necessary introduction. Program managers will also be interested in the principles and processes of VV&A as they incorporate these into their programs. Finally, Chapter 5 is important to assist program managers in preparing senior decision makers for the accreditation decision.

The technical staff whose job is to do the actual V&V should read the entire document. In addition to the chapters noted above, Chapter 4 will give these users valuable guidance on specific techniques that are used in V&V, and Chapter 6 will provide common reporting formats to help them document the VV&A effort.

Chapter 1 — Overview

1.1 Preface

This document provides you, the DoD program manager or M&S (Modeling and Simulation) manager with an understanding of basic Verification, Validation, and Accreditation (VV&A) terminology and techniques. Its goal is to help you develop an informed and independent judgment about how credibly models and simulations (M&S) are being integrated into your program. To understand why you should be concerned with the material in this document, imagine yourself in the following situation.

> *You are a senior officer or civil servant working for one of the Services. You have just been tasked to provide a comprehensive solution to a major military problem. That problem may be the development of a new weapon system, the design of a training exercise, or perhaps the definition of military force structure requirements in your branch of the Service for the next three decades. You have little time, less money, and only meager human resources to complete the task. You know (or you have heard) that one of the ways to save time, money, and human resources is to take advantage of the breathtaking array of models and simulations that have been made possible by the dramatic increase of computer hardware and software capability in the last decade. You don't know much about M&S and perhaps still less about particular simulations, but you know how to get to the people who do. So you set up an M&S shop within your organization; you allocate precious resources to a staff of analysts, scientists, engineers, and warfighters; and you charge them with the delicate task of pulling together a credible M&S effort that will meet or support key program objectives while saving time and money. You figure if these people can't do it, nobody can. You walk away happy.*
>
> *Time passes. Things go along pretty well for a while, or at least they appear to. Every so often you call for a program review that includes the status of M&S efforts. "Everything's fine," you're told. The M&S suite has been selected and stabilized, M&S outputs have been related to key*

program Measures of Merit (MOMs), M&S reviews have been scheduled and conducted, and your M&S shop is confident that their results are credible on the basis of "VV&A."[1] "VV&A?" you ask. "What's that?" Your M&S team throws alarmingly technical terms around that make it sound as if big money is being spent. "Not to worry," they say. The "industry" has been doing "VV&A" for years; the Military Operations Research Society (MORS) has standard definitions for key technical terms and techniques; and no unnecessary "V&V" is being done. The models and simulations supporting this program will be "VV&A'ed" in time to meet major program milestones. You walk away happy. Sort of.

You do a little research. You discover that, far from being a compact, tightly knit, well-defined discipline, VV&A spans a broad spectrum of activities. You discover that the depth and breadth of these activities depend not only on the kind of M&S to which they are applied but also to the specific application for which the M&S will be used. You discover that "community consensus" about the definitions of verification, validation, and accreditation exists at only the most general level. You also find out that the definition of the V&V techniques that should be used for specific types of models and simulations and how these techniques should be applied to establish the credibility of M&S when used for particular applications is a subject of intense debate. You discover that a major high-level review of your program is fast approaching, and you suspect that some questions about all this VV&A business will come up because of the attention given it in recent DoD and Service policy documents. You wish you knew how to make an independent judgment of how well your M&S team has met its critical milestones to support your program's objectives. You walk away maybe not so happy.

Sound familiar? Then this document is for you.

The information in this document has been compiled from a wide variety of sources, including recent DoD Directives and Instructions related to M&S management and VV&A; software industry standards and practices; the practical experience of numerous ongoing VV&A efforts across the DoD and industry; academic texts and professional literature; and professional societies and organizations intimately familiar with M&S and VV&A. The hope is that this broad array of experience, concisely presented, will encourage you to pursue VV&A of M&S with confidence, vigor, and insight.

In addition to this introductory section, this chapter consists of six sections that provide (a) an understanding of basic V&V techniques and terminology (Section 1.2); (b) an appreciation of the value of VV&A (Section 1.3); (c) a discussion of where VV&A fits in the scheme of M&S (Section 1.4); (d) a discussion of limitations to VV&A (Section 1.5); (e) a general introduction (Section 1.6) to some practical aspects of VV&A, such as tailoring V&V tasks to the requirements of your specific application, who should be doing what (and why), and costing and scheduling considerations; and (f) a description of the rest of this Guide (Section 1.7).

1.2 What Is VV&A?

This section defines these terms: model, simulation, simulator, M&S, verification, validation, accreditation, and other related terms.

M&S credibility is measured by verification and validation (V&V) and formally approved as adequate for use in a particular application by accreditation. The entire process is known as VV&A. Before we define the individual elements of VV&A, let's get a few preliminary terms out of the way.

1.2.1 Terminology

One of the most confusing aspects of M&S terminology is the difference between a model and simulation. In fact, many people in the M&S community either do not really know (or do not really distinguish) between the two in conversation. In fact, there is no official consensus as to the definitions of these terms, nor do we propose to settle the debate within the context of this venue. The general distinction between a model and a simulation will be important, however, when we talk about the details of VV&A. We have developed an approach to explaining the terminology, therefore, that is consistent with (most) current definitions, has practical utility, and is not illogical.

According to DoDD 5000.59, a *model* is "a physical, mathematical, or otherwise logical representation of a system, entity, phenomenon, or process." A *simulation* is "a method for implementing a model over time." But what do these definitions mean in lay terms? And how does the distinction between them affect the nature of VV&A activities?

A *model* is a conceptualization, an abstraction of some physical phenomenon or process into mathematical equations and solution approaches (called "algorithms"), each with its own assumptions, limitations, and approximations. For example, the radar range equation is a model, an abstraction of the radar detection phenomenon into an equation that makes certain assumptions about how radar energy interacts with targets, clutter, and the atmosphere. If you take this equation and convert it into a computer program (software) to solve it for particular scenarios as a function of time (say, to determine the detection history of a combat aircraft during a mission from a fixed radar site), the result is a *simulation*, which is a software framework that executes a model (or models or model pieces) in the proper order, provides timing and coordination between them, and controls the inputs and outputs. Thus, a model is an abstraction of a phenomenon into mathematical equations and algorithms, whereas a simulation is the software implementation and solution of those equations and algorithms over time within the context of a scenario. A model can exist without a single piece of software; a simulation is the software that implements the model over time.

Another potential point of confusion is that many people in the M&S community use the term *M&S* to stand for both *models and simulations* and *modeling and simulation*. *Modeling and Simulation* is an analytical problem-solving approach; *Models and Simulations* are mathematical abstractions and software implementations. Although the community uses the terms interchangeably, this document distinguishes between the two.

And, lest this topic become too easy to understand, we introduce yet another commonly used term that may cause the newcomer to M&S terminology some confusion: *simulator*. In its broadest sense, a simulator is a training device made up of some combination of hardware and software designed to provide an artificial (but suitably realistic) environment in which a human player can interact with those aspects of reality in which training is desired and within which all aspects of reality that are simulated interact realistically with each other. Flight training simulators come to mind as good examples. Not all aspects of reality need to be simulated in a simulator, only those crucial to the goal of training. Similarly, not all aspects of reality that are simulated need to be simulated with complete realism, only enough to ensure that training objectives are met.[3]

Simulators also can be used for testing, but here their required level of realism typically is greater. The most obvious case of simulators built for both training and testing applications are the open-air threat missile system simulators found on many DoD test ranges. These systems are used not only to train pilots in the proper use of available equipment and combat tactics but also to test the effectiveness of new electronic countermeasures (ECM) systems designs. With these simulators, the environment created is not enclosed (as it is in the case of a flight simulator), but the simulator still reproduces those aspects of reality essential to the training or testing application (e.g., a realistic, open-air RF environment).

One final point concerns the distinction between the terms *verification* and *validation*. Most people have an intuitive understanding of the meaning of the term "validation" with respect to M&S. Ask them to tell you the difference between verification and validation, however, and you're likely to get a blank stare, because these two words have the same or similar meanings to many people. To the M&S community, however, there are important distinctions.

The distinction between the two is most easily remembered in terms of their focus. At the risk of oversimplification, verification focuses on M&S *capability*, whereas validation focuses on M&S *credibility*. *Verification* ensures that a simulation meets all the requirements specified by the user and that it implements those requirements correctly in software; *validation* ensures that a simulation conforms to a specified level of accuracy when its outputs are compared to some aspect of the real world. We'll explore the nuances associated with determining the level of accuracy required of a simulation later. For now, just realize that verification and validation add separate, distinct, and essential kinds of credibility to M&S. Neither achieves its fullest contribution to M&S credibility without the other.

With basic definitions and distinctions out of the way, let us now turn to more detailed descriptions of verification and validation (V&V).

1.2.2 Verification Basics

According to DoDD 5000.59, *verification* is "the process of determining that a model implementation accurately represents the developer's conceptual description and specifications." In more colloquial terms, verification is the process of determining that a *model* and its resultant *simulation* (remember our definitions) accurately represent both what is required and what the M&S developer says will be built for you in accordance with those requirements.

If you are planning on developing models or simulations for use in your program, you need to do two things before a single line of software (usually referred to as *code*) is written. You need to build and verify a *conceptual model* from which the code will be written (Conceptual Model Verification), and you need to verify the proposed design that will support development of the simulation's code (Design Verification). A mapping of the proposed design elements back to the conceptual model and your M&S requirements helps to document that your requirements are appropriately addressed and that there is traceability between those requirements and the proposed design.

Before you can verify a conceptual model, you have to have one. In the ideal world, simulation development would not proceed until the underlying M&S requirements were fully identified on the basis of the requirements of the problem at hand and until a fully verified conceptual model was developed from these requirements. In the real world, of course, we all know that M&S development usually proceeds with inadequately defined or rapidly changing requirements. It is very important, however, that you not sacrifice accuracy on the altar of expediency. Take the time to identify your simulation requirements in as much detail as possible early on. Do so by defining your problem concisely and accurately; by defining the simulation outputs, functions, and interactions that will be required to answer your problem; and by specifying, at least in general terms, how much like the real world you need these outputs, functions, and interactions to be. (See Section 1.6.1 for more details.) The developer will then take these requirements and produce a conceptual model.

A *conceptual model* is a simulation developer's way of translating your modeling requirements into a detailed design framework, from which the software that will make up the simulation can be built. A conceptual model typically consists of a description of how your modeling requirements were broken down into model-able pieces, how those pieces fit together and interact, and how they work together to meet the requirements you specified. It also should include a description of the equations and algorithms that will be used to meet your requirements, as well as an explicit description of any assumptions or limitations made or associated with the equations, algorithms, or solution approaches that were used to solve your modeling problem. The conceptual model also should identify how these assumptions and limitations might impact the simulation's ability to meet your requirements, once it is built. The process of reviewing the conceptual model and ensuring that it meets your specified requirements is called *Conceptual Model Verification.*

After the conceptual model is verified, the developer produces a *software design specification*, which describes exactly how the conceptual model will be translated into software. It defines the components, elements, functions, and specifications that will be used to produce the simulation's software based on the conceptual model. The process of reviewing the detailed design to be sure it conforms to the conceptual model is called *Design Verification*.

Once verified, the conceptual model and its associated design are converted into actual software by the developer. At this point, you have one last verification hurdle to overcome: verification of the software itself (usually called *Code Verification*). Code verification guarantees that the detailed design is implemented correctly in the software. Code verification normally entails detailed desk checking and software testing of the code, comparing it to the design elements, specifications, and operational criteria that were approved during verification of the conceptual model and detailed design, documenting any discrepancies and fixing any problems discovered.

What if you're not building a new simulation, but just want to use an existing one "off the shelf"? How can you determine that the conceptual model and design specifications of this simulation (over which you had no developmental control) meet your M&S requirements? Before we discuss this, let's define what we mean by *off the shelf*.

Most of the models and simulations in this category are called *legacy* M&S because they have some history of prior use. In addition, some legacy models and simulations in wide use were built before the advent and widespread implementation of detailed software design standards and practices. This does not necessarily mean that they are badly designed (although they certainly can be). A good legacy simulation is characterized by a long history of consistent use and development by an active (usually large) user group, good configuration management and documentation, and widely recognized community acceptance of its results.[3] The most important thing that legacy models and simulations may not have that more recent ones do (or, least, should) have, is detailed documentation of their conceptual models and the design specifications that flow from it. Models and simulations without such documentation may require that a suitable substitute for the conceptual model be generated from an analysis of the code as it currently exists and from any available documentation. Once the conceptual model and existing design elements have been identified and documented, however, you still need to determine if the result meets your M&S requirements. Because you had no control over the conceptual model (or the design requirements and specifications) of a legacy model and simulation, the usual verification of the conceptual model and its associated design may not be appropriate. What you can do, however, is review and compare the legacy simulation's assumptions, limitations, and design elements to your M&S requirements to evaluate whether the simulation as it stands meets your requirements. This is called *Conceptual Model Validation* (see below).

It should be clear from the previous discussion that verification requires a clear understanding between you and the simulation developer about your M&S requirements and about the developer's interpretation (and implementation) of those requirements. This understanding and agreement drives the conceptual model, the simulation design and development based on that model, and your ultimate assessment of the simulation's suitability for your application. *Clear requirements and specifications are crucial to cost-effective verification efforts.*

A number of well-established techniques that can be used for verification are discussed in Chapter 4.

1.2.3 Validation Basics

According to DoDD 5000.59, validation is "the process of determining the degree to which a model is an accurate representation of the real world *from the perspective of the intended uses of the model.*" Notice the emphasis. It is critical that the simulation be assessed in terms of how it will be used. Accurate knowledge of how the simulation will be used determines the degree of detail that must be represented for the simulation to provide usable results and the degree of correspondence with real-world phenomena that will be sufficient for you to use the simulation with confidence. The less you really know about how a simulation will be used to solve your problem, the more likely it is that you will have to over-specify validation requirements "just in case."

Thus, there are two prerequisites for cost-effective validation: a clear understanding of the intended uses of the model, because this sets your requirements for functionality (i.e., what needs to be modeled) and for fidelity (i.e., how well those functions need to match the real world) and a clear definition of the real world. If you don't have a good definition of what you're validating against, you won't be able to determine the difference between a good validation result and a bad validation result. For example, will you validate a simulation against range data, laboratory data, another simulation, or the opinion of experts in the field? Each of these real worlds has inherent drawbacks and limitations that can make or break the apparent validity of a simulation.

In its simplest form, validation consists of comparing a prediction (from a simulation) with an observation (from the real world), and making a judgment about whether the result is good enough for application to your problem. Simple as this concept is, validation techniques are not limited to comparison of simulation results with test data. They also may include sensitivity analyses to test simulation performance against extreme conditions, comparison with other models and simulations known (or assumed) to have validity in the operating range required, and the opinion of subject matter expert (SME) reviews of M&S results.

Validation typically is addressed at two levels: conceptual model validation and results validation. *Conceptual Model Validation* is the determination (usually by a group of SMEs) that the assumptions underlying the proposed conceptual model are correct and that the proposed simulation design elements and structure (i.e., the simulation's functions, their interactions, and outputs) likely will lead to results realistic enough to meet the requirements of the application. The difference between conceptual model validation and conceptual model verification is a subtle but important one. Conceptual model *verification* ensures that the proposed conceptual model (and its resultant design) satisfies the *functional, interactional, and output* requirements imposed by the specifics of your problem; conceptual model *validation* ensures that the proposed conceptual model (and its resultant design) satisfies the *fidelity, accuracy, or credibility* requirements imposed by the specifics of your problem. The difference is most easily colloquialized as the difference between the questions "Did I build the thing right?" and "Did I build the right thing?"

Results validation compares the responses of the simulation with known or expected behavior from the subject it represents to ascertain that those responses are sufficiently accurate for the range of intended uses of the simulation. This process includes comparison of simulation outputs with the results of controlled tests, sensitivity analyses, or expert opinion.

An important aspect of validation to remember is that validation will not say a simulation is good or bad. It simply measures the difference between simulation outputs and the real world. The user then decides if that difference is small enough for the simulation to be used in a specific application and if the results when used in that application will have the expected accuracy. (More about this in the next section.)

One final observation on validation. Most simulations are composed of thousands of lines of computer code or thousands of electronic circuits and components (or both). The logic diagram of the alternative paths through a typical simulation is extremely large: sufficiently large, in fact, that it is, in practice, impossible to check every possible path. Hence, for all practical purposes, a simulation cannot be completely validated. Therefore, for the question, "Is this simulation validated?" the answer should always be, "Yes, for the conditions specified in the validation report." Validation is performed on those aspects of a simulation that are important to a particular application. This makes validation feasible and provides the measures of fidelity in areas most important to successful simulation results.

Some of the more common validation techniques and methods are discussed in Chapter 4.

1.2.4 Accreditation Basics

Once a simulation has been verified and validated[4] in accordance with requirements defined by the intended application, an official statement that it is acceptable for the specified use must be made. According to DoDD 5000.59, *accreditation* is "the official certification that a model or simulation is acceptable for use for a specific application." In many cases, *Expert Review* is the process used to evaluate V&V results in light of M&S requirements defined by the specifics of the problem. These reviews identify credibility gaps, assess their risk to the program, and make recommendations for (or against) accreditation of specific models and simulations.

The accreditation agent (e.g., a program manager) should participate in the earliest stages of M&S development to become familiar with M&S requirements and acceptance criteria and to identify expert review requirements and appropriate SMEs as early as possible. Early involvement helps mitigate the risk of executing an M&S program that will not meet overall program requirements for M&S credibility. In the final stages of the V&V program, the accreditation agent should participate in the summary evaluation of any V&V results and supplemental M&S information to ascertain the adequacy of M&S efforts and the readiness of the M&S suite for final accreditation.

It is important to recognize that accreditation is not (or, at least, should not be considered) a foregone or assumed conclusion. It is a decision that a specific simulation can be used for specific application, based on objective evidence of suitability for the application. Hence, a simulation can receive an accreditation for use in one specific application (e.g., a flight training application) but not be accredited for use in another specific application (e.g., aircraft system design in an acquisition program).

A process leads up to an accreditation decision. This process gathers all the information about specific model or simulation capabilities relative to the requirements of a specific application. This information includes verification and validation results but also includes such things as simulation run time, number of simulation operators required, the simulation's history of use, documentation status, configuration management, and other factors that will be discussed in Chapter 5.

1.3 Why Do VV&A?

This section offers six reasons why VV&A is a good idea. It's worth spending a little time dispelling some common misconceptions about the value (or lack thereof) of VV&A. Why all the fuss, anyway? Isn't VV&A just another check in the box, added to an already lengthy list of such boxes?

In a word, "No."

This section will discuss six benefits of VV&A:

- Increased confidence in M&S use
- Reduced risk of M&S use
- Increased M&S usability for future applications
- Cost containment
- Potential for better analysis
- Satisfaction of policy requirements

Although by no means an exhaustive list of potential benefits, these have the most impact.

1.3.1 Increased Confidence in M&S Use

A well thought-out program of V&V activities tailored to the application for which a simulation will be used does much to establish or improve confidence in the use of that simulation for that application. V&V increases confidence in models and simulations by providing objective evidence of credibility *within the confines of that intended use*. Notice the emphasis. V&V, by itself, does little to increase confidence in M&S use unless application-specific requirements for credibility are developed and defined for that use. The challenge to the V&V practitioner, therefore, lies in the selection and scoping of that set of V&V tasks most appropriate to the application at hand. Credible tailoring of V&V activities to specific applications, in turn, requires a clear understanding of the contribution that each V&V technique makes to the credibility of M&S and a knowledge of the M&S functions that are critical to the problem at hand.

Chapter 4 defines V&V techniques and their contributions to M&S credibility for specific classes of models and simulations and their applications. Section 1.6.1 discusses tailoring schemes that allow V&V practitioners to focus V&V tasks on the particular requirements of an application to minimize VV&A cost and schedule. As a general rule, however, it is safe to say that the V&V techniques that lend the most credibility to M&S use are not those that cost the most. In particular, V&V status reports and M&S usage histories can help to reduce the scope of new V&V efforts and to indicate the range of applications for which M&S results have been considered acceptable for use. The cost of this aspect of V&V is much less than the detailed code verification and validation with large amounts of test data envisioned by most users when they think of V&V. A history of prior accreditations also lends considerable weight to the choice of models or simulations for a given application by establishing the degree to which M&S results have been considered acceptable by prior users for similar applications. Again, the cost of an accreditation history review is negligible compared to performing more detailed V&V.

1.3.2 Reduced Risk of M&S Use

A major corollary of increased confidence in M&S use is the reduced risk of relying on models and simulations to support major program decisions, objectives, and milestones. Incorrect or inadequate M&S can lead to corrupted system concepts and requirements, poor system design, inaccurate results, negative training, and even system failure, possibly with catastrophic loss.[5] V&V reduces the risk that M&S use will lead to incorrect or indefensible results. The issue in this case is not really "What is the cost of V&V?" but rather "What is the cost of NOT doing V&V?" What is the cost, in terms of time and money, of making an incorrect decision based on M&S results? These hidden costs of avoiding V&V are frequently intangible, unpredictable, and unquantifiable. As a result, they tend to be ignored in the calculation of the value added by V&V. Nevertheless, reduced risk in using M&S is a major benefit of performing V&V *tailored to the application*.

1.3.3 Increased M&S Usability for Future Applications

The requirement to perform V&V to establish the credibility of M&S for use in DoD applications establishes a beneficial dynamic that can reduce the long-term cost of both M&S use and V&V. This is because V&V activities performed by multiple users on a stable simulation, typically one with a well-defined configuration management and development policy (see Section 1.6.2), will, over time, establish a body of evidence supporting its credible use for a wide variety of applications. Different users will, of course, focus their attention on different aspects of V&V to support their individual applications; outside of your program, you have no control over the V&V that gets done. But as the V&V sample space for a specific simulation grows and with it, the body of evidence supporting its credibility, the more likely that it will receive more development and V&V attention. Other models and simulations that perform similar functions but that do not fare well in V&V or that do not have a V&V pedigree adequate to support credible use will give way to those that do. In this way, V&V becomes a natural selection process for the development of fewer models and simulations but with greater capability and established credibility. From this standpoint, your program benefits from the V&V of others for common-use models and simulations. The same dynamic is likely to apply within your own program, meaning that other programs will benefit from your V&V of a particular simulation, just as you benefit from the V&V of others.

Reducing the duplication and improving the credibility of DoD models and simulations may not number among the proximate goals of the typical program manager when V&V is performed. It is clear, however, that the net effect of V&V activity across a spectrum of users of individual models and simulations will be to improve both their capability and their credibility over time.

1.3.4 Cost Containment

If V&V results are documented in a standardized way (see Chapter 6) and if these results are made readily available to the user community, the cost of V&V to support accreditation will drop. New accreditation efforts can build on the V&V results of earlier users. In this way, improvement of the credibility of individual models and simulations becomes a bootstrap process, with multiple users contributing to the body of knowledge about the simulation. This common body of evidence eventually benefits all users of the simulation.

A beneficial consequence of consolidating V&V results across a M&S user community is that V&V becomes market-driven, reducing the duplication of V&V activities. When individual users have to retrace V&V ground that may have been covered by others, the efficiency of overall V&V efforts for the simulation is reduced. But when a consolidated body of V&V knowledge exists, users can focus on the areas of the simulation that need the most attention for their particular application. The analytical needs of a simulation's user community can thus drive the depth to which V&V data are collected, and individual users (like you) in the community no longer waste precious V&V dollars chasing V&V products that already exist.

This assumes, of course, the existence of standard V&V processes and products within individual M&S communities and ready access to this information by individual members of these communities. DMSO is encouraging ready access to V&V information via the MSRR. In this way, both prerequisites for cost-efficient V&V to support accreditation for diverse M&S communities are being met. Your V&V efforts contribute to the body of knowledge about individual models and simulations, and that contribution benefits all users.

1.3.5 Better Analysis

Before widespread use of M&S, effective problem-solving required the clear definition of the problem and its solution objectives, the charting of the analysis with flow diagrams, and the development of an outline of the expected results. With the advent of complex computer simulations that have great predictive power, however, much of the discipline attached to the analytical process has been neglected in favor of understanding the simulation itself. There has been a growing tendency, for example, to focus analytic efforts on gathering valid input data for simulations (see Section 1.6.5), and on taking advantage of the expanded scope of analysis afforded by high-power computers by running a multitude of simulation cases. In essence, analytical depth is being sacrificed for breadth. Rather than being used to do better analysis, models and simulations are being used to do more analysis.

The requirement to perform V&V, however, coupled with the necessity of narrowing its scope to contain costs, can provide an incentive to rejuvenate sound analytical practices within your program. Cost-effective V&V requires the development of detailed M&S requirements that are focused on the intended use of particular models and simulations for particular applications. Development of these requirements necessitates the clear description and full characterization of the analytical problem and approach to identify required information elements, derive appropriate metrics, identify analytical constraints, determine appropriate M&S outputs, and, in general, integrate M&S into your program in a credible way. The discipline required to develop well-defined M&S requirements clarifies analytical issues and facilitates the development of more thoughtful analytic techniques and approaches. Thus, the requirement for cost-effective V&V requires a return to the basic practices of analytical problem-solving that have fallen into disuse. The result can be a tendency to improve the quality of the analysis applied to your program.

This is not to suggest that VV&A automatically leads to better analysis. Improperly done, VV&A can actually detract from simulation credibility by making it appear that critical credibility issues have been addressed adequately, when in fact they have been improperly addressed. It is the synergism and interplay between VV&A and analysis that, when properly managed, can lead to improved confidence in the results of analysis using M&S.

1.3.6 Satisfaction of Policy Requirements

If you're still not totally convinced of the value of VV&A, there is one more argument that might turn the trick. We've held it until last because, although it's a persuasive argument, it's not very popular, and it certainly isn't intellectually satisfying. Simply put, you don't have much of a choice.

The inescapable facts are these: (a) M&S will be used more and more across DoD (and industry) to save time, money, and resources, and (b) people in *very* high places are *very* worried about how M&S, both new and old, can be integrated into DoD applications in a credible, justifiable, cost-effective way.[6] This means that, like it or not, VV&A will probably play an increasingly influential role in every aspect of DoD operations that contains M&S. And M&S is playing a greater role in every aspect of DoD operations. It's as simple as that.

1.3.7 Benefits Summary

Although the requirement for VV&A of your M&S is going to get harder to address in the coming years, you should have some appreciation by now of why VV&A is worth addressing in the first place. In short, VV&A

- increases the objective confidence you have in your M&S program
- reduces the risk of making the wrong (possibly catastrophic) decision for a critical study, exercise, or acquisition based on incorrect M&S results
- reduces the proliferation of M&S within your program and focuses V&V attention on those models and simulations most useful to your problem
- results can be leveraged to reduce future VV&A costs
- can require the M&S and Analysis shops within your program to focus more on sound analytical practices in order to define the most cost-effective V&V program that meets your requirements for M&S credibility
- meets Service and DoD policy requirements while preserving technical merit

1.4 Where Does VV&A Fit in the Scheme of M&S?

This section offers an overview of VV&A's place in model development and use. It provides a larger context for M&S use in an application.

It should be remembered that M&S is simply a tool or technique that can be used to solve a problem and that VV&A is just a way to gain assurance that the selected model or simulation can produce meaningful results relative to the problem's solution. The problem that needs to be solved is usually called the *application*. The process for solving the problem is usually referred to as the *application process*. The application process context for VV&A and M&S is shown in Figure 1-1.

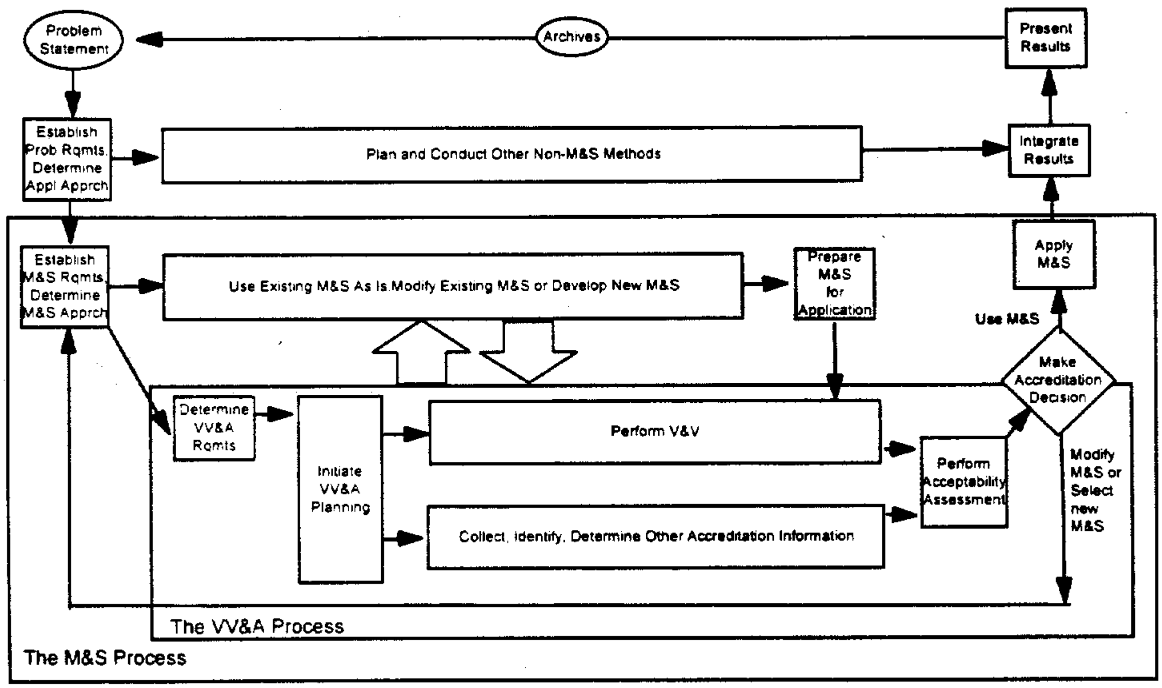

Figure 1-1. VV&A in the Application Process (Scheme of Things)

The application process begins with a clear and unambiguous statement or definition of the problem. A good definition of the problem makes it easier to define its solution requirements. These requirements are the features, characteristics, or functions that are important to the problem and essential to its solution. For example, if the need is to develop a new ECM system, it is essential to define the expected combat systems on which the ECM system will be hosted, the threats against which the ECM system is expected to work, the required effectiveness of the ECM system, the operational environment in which the ECM system will operate, and the other systems expected to be in the operational environment. Measures of Effectiveness (or Merit) that will determine if the requirements have been met are derived from these characteristics. Methods or ways of producing values for these measures or addressing the requirements directly then are determined. These methods can include research into work already accomplished, design analysis, direct testing, or M&S. A complex problem usually employs a number of these types of methods to achieve a robust solution. The set of methods that addresses all the problem requirements is integrated into a consistent, logical application approach.

The application approach shows the problem requirements that will be satisfied by specific methods and the measures that will be used to evaluate the success of each method in fulfilling the solution requirements of the problem. Those methods unrelated to M&S are planned and executed. The requirements to be satisfied by M&S are identified separately and form the basis of the M&S approach. As part of an initial M&S approach, the types of models and simulations that can be used are identified, as well as the criteria for determining when a model or simulation is acceptable for this application. The specific model(s) and simulation(s) to be used for this application are selected according to these criteria. The VV&A status of a model or simulation can be a factor in M&S selection. For complex applications, a number of models and simulations may be necessary to satisfy the M&S requirements of the problem. The M&S approach may call for using specific models or simulations as they are, modifying existing models and simulations, or developing new models or simulations.

Once the M&S suite has been selected and the M&S approach finalized, work can begin on establishing, modifying, or developing the model or simulation. The VV&A process begins immediately and uses the M&S requirements, the acceptability criteria, and the VV&A status of the selected models and simulations to determine the VV&A requirements for this application. Based on these requirements, a plan to accomplish the necessary V&V is developed. Although V&V will produce significant information about the model's or simulation's capability to support the application, additional information beyond V&V is also useful. This other information can include the model's or simulation's configuration management status, documentation status, previous use in other similar applications, and development standards used. This other information (and the V&V results) is a factor in the acceptability assessment. The acceptability assessment compares the model's or simulation's capabilities and limitations to the acceptability criteria and assesses overall its acceptability for this application. This accreditation assessment report includes a recommendation whether to accredit the model or simulation, along with the rationale for that recommendation.

This technical assessment then is given to the accreditation authority, who must decide, using the assessment information provided, whether the M&S suite is acceptable for use in the application. The decision may be to use the M&S suite as it is, to limit the use of the results of the model or simulation, to perform (additional) modifications to the model or simulation, to perform additional V&V, or to reject the M&S suite completely for this application.

If the decision is to use the model or simulation, the M&S runs and exercises are performed. The results are integrated with the non-M&S results to solve the problem. Archiving the results of the VV&A activity in the appropriate MSRR for future use is important. Any V&V carried out for this application will reduce the amount of V&V that may be necessary for those models and simulations in future applications.

Chapter 5 discusses in more detail the role of VV&A in the context of application problem-solving.

1.5 Common Misperceptions About VV&A

This section describes the limitations of VV&A. It also explains why each new application must be accredited and why V&V must be reviewed (and possibly repeated or expanded) when a model changes. Three common misperceptions about VV&A arise from a misunderstanding of the nature and value of VV&A.

1.5.1 VV&A Is No Substitute for Sound Analysis

VV&A enhances a simulation's credibility and reduces the risk of its use in a particular application, but VV&A cannot guarantee that the M&S results will be correct, that the results will be correctly analyzed and interpreted, or that the right model was chosen to solve the problem. It can identify a model's weaknesses, but the correction of the weaknesses or their workarounds is not a part of the VV&A process. If the M&S requirements or acceptability criteria are incorrect or ill-defined, the likelihood that an incorrect M&S may be selected and used increases. The VV&A process will not assess the correctness of the M&S requirements or acceptability criteria.

The quality of the VV&A process used to support an application also depends on the thoroughness of the VV&A effort and on the capability and experience of the VV&A team. Unfortunately for some applications, VV&A is done in an afternoon meeting of project team analysts who have limited knowledge of specific M&S and application requirements. The results of this kind of VV&A create a higher risk of poor integration of M&S into problem-solving. You get what you pay for.

1.5.2 Accreditation Is Not a One-Size-Fits-All Check in the Box

Accreditation is a decision to use a *specific* simulation for a *specific* application. Each application has a different set of requirements and detailed acceptability criteria. No two problems are exactly alike. V&V can be done without detailed knowledge of the values of simulation acceptability criteria, but accreditation cannot be performed without application-specific requirements and detailed acceptability criteria.

Moreover, when a simulation is modified, it is usually modified to improve its operation, simulation accuracy, or simulation scope. These changes may affect the simulation's suitability for particular applications. The changes to the simulation must be compared with the modeler's intent (verification), and the impact of the changes on simulation output also must be compared with the real-world system or process to measure the increase or decrease in fidelity (validation). Additionally, when the real-world changes or the model or simulation is used for a purpose different from the original intent, previous VV&A results should be reviewed to determine the impact of these changes on the credibility of the simulation. Because the real world is rarely static over any length of time, it is useful to review a model's or simulation's VV&A status periodically to ensure consistency with the current projection of the real world.

The practical impact of all this is that VV&A cannot be considered a solitary task. Although much of the groundwork for accreditation will remain fixed once the basic information is documented during development V&V, accreditation for specific applications (and after simulation changes) is still necessary.

1.5.3 VV&A Is Never Completed

This misperception is really a corollary of the previous one. Many M&S users are surprised when the issue of VV&A activities arises after development or initial accreditation. If you're tempted to say, "I thought we did all that," you have fallen victim to the most common misperception about VV&A.

VV&A is never finished because simulations cannot be verified or validated completely. Complete verification requires testing of every logical branch and condition of the simulation under all possible combinations of input parameters. Complete validation requires comparison of every possible set of input conditions to data run under identical conditions in the real world. It doesn't take a very complex simulation to exceed the number of practically attainable software tests or testable validation conditions.

This does *not* mean, however, that VV&A is an unattainable Holy Grail; it means only that you should expect VV&A activities to continue throughout the life cycle of M&S development *and* application to particular problems. The scope of VV&A required to establish M&S credibility for any particular problem always will be manageable and determined by the specifics of the problem. Ongoing VV&A activities are the price you should expect to pay for ascertaining and maintaining the credibility of your models and simulations.

1.6 Some Practical Considerations

Right now you're probably pretty nervous. VV&A is not just a check in the box, and (oh, by the way) you're never done. The following sections discuss some practical considerations that should put your mind at ease. In particular, we discuss scoping VV&A efforts to meet your requirements; key players, roles, and functions in VV&A to help you organize your efforts; and configuration management issues that help keep track of VV&A activities and relate them to your particular stage of M&S development. All of these topics will help you extend the shelf life of VV&A results.

1.6.1 Scoping and Cost

This section offers guidelines for estimating VV&A needs based on the application type, its importance, and previous VV&A activity. It also discusses how much VV&A is enough.

Right now, you're probably wondering, "What is all this going to cost me? I have heard that software V&V can consume 25–30 percent of my M&S development budget. I don't have 25–30 percent of my budget to devote to *anything*."

Your well-founded concern reflects the recent focus on the credibility of M&S, which has been balanced by an equal concern for the cost of the V&V activities that contribute to it. The M&S community lacks a coherent process that links V&V information to application-specific requirements for M&S credibility. This lack has prevented M&S users (like you) from identifying cost-efficient sets of V&V activities that meet credibility requirements for individual applications. The natural result has been a tendency to overestimate V&V requirements, with the corresponding (mis)perception that "V&V costs too much and takes too long." Operating under this misperception, cost and schedule pressures can lead easily to an irresistible temptation to dilute M&S credibility requirements to meet fixed (usually meager) V&V budgets. The end result leans toward accreditation by fiat, rather than by objective evidence. What's a program manager to do?

1.6.1.1 Exorcising the Cost Demon

First of all, don't be misled by what appear to be overblown estimates of the cost of VV&A. There is a great deal of misinformation on the exaggerated cost of VV&A propagated by people who have little or no first-hand experience in performing it or who have a vested interest in ensuring business continues to be done as usual. The overwhelming evidence from a large number of samples indicates that costs have been well-controlled and tend to cluster or correlate in a predictable manner. Historical data show, for example, that the percentage of M&S development funds devoted to the assessment of M&S credibility spans a reasonably narrow spectrum, from a low of about 5 percent to a high of about 17.5 percent, with most efforts somewhere in the middle range of 10 to 12 percent.

Even these costs depend somewhat on the aspects of VV&A that are included in the estimate. Some think that all V&V and test and evaluation (T&E) activities performed by the developer should be considered part of the total cost of VV&A, leading to the anecdotal estimates of 25–30 percent of development costs, whereas others count only those activities specifically required to accredit a simulation for a given application, leading to estimates closer to half of the previous ones. Either way, the historical record shows that the high estimates tend to include V&V tasks not necessarily essential for M&S accreditation, whereas the minimum levels tend to be a bit Spartan and may not always provide the full range of V&V data necessary to make a strong case for M&S credibility. As in all things, moderation is the key.

1.6.1.2 Trading Off Cost Against Credibility or Risk Reduction

But what constitutes effective moderation? In estimating the costs of your V&V efforts, should you stay closer to 5 percent or 17.5 percent of your M&S budget, or should you just shoot for the average (11.25 percent) and live with the results? How can you tell whether or not the V&V activities you buy for *any* amount of your budget will meet your M&S credibility requirements?

First of all, you'll have to accept that selection of V&V activities on a fixed budget will always involve a trade-off of cost against credibility. Truly cost-effective VV&A seeks to balance the requirement for M&S credibility and risk reduction, driven by the specifics of your application, with real-world constraints, driven by the program M&S budget. Final selection of the exact set of VV&A activities depends strongly on the defined needs, known problem areas, and high-risk aspects of your program, as well as on the availability of tools, methods, human resources, and facilities. When done in good faith, however, VV&A has been shown to provide more in benefits than it costs in resources. It is unquestionably an added-value process, but V&V activities must be chosen correctly. The real question is not, "How much should I spend?" but "What should I buy?"

Guidance from the M&S professional community on how to select the most cost-effective set of V&V tasks to meet a particular requirement for credibility has matured in recent years.[7] The process of selecting V&V tasks rationally within a constrained budget involves answering three key questions about the integration of M&S in your program: What do you need M&S to do? How well do you need M&S to do it? How well do candidate models and simulations do what's needed? If you can answer these three questions, you can select a cost-efficient set of V&V activities that meet your requirements for M&S credibility. Most M&S experts would agree that faithful execution of two activities contributes greatly to the development of a well-focused (hence cost-effective) V&V program.

1.6.1.2.1 Application Analysis. First conduct an in-depth analysis of your problem to define what you want M&S to do. Before *any* decisions about applying M&S to a given problem are made, the problem itself must be defined and articulated clearly enough to see *where* models and simulations help solve the problem and *how* they will help solve the problem. *An ill-defined problem is the most common reason for failure to integrate M&S credibly into program objectives.* A sound problem analysis consists of four elements: (a) a correlation of *clearly articulated* program objectives with the decisions that must be made to reach those objectives, similar to a decision hierarchy or tree; (b) development of a *well-defined* set of Measures of Merit (MOMs)[8] by which each decision will be addressed and resolved; (c) an identification of the program decisions and their associated MOMs that will be addressed, resolved, or supported by M&S; and (d) an identification of the required predictive capabilities that models or simulations must have to support each program decision, i.e., M&S functional requirements. The correlation of program objectives, decisions, MOMs, and M&S functional requirements is the single most important aspect of the V&V tailoring process, because it forms a template for the integration of M&S into your program.

1.6.1.2.2 Acceptance Criteria Definition. Next, develop acceptance criteria for models and simulations you might want to use in your program. Having defined *what* M&S will be required to do (the functional requirements), it remains to determine how *well* candidate models and simulations must do them. The answer lies in two types of acceptance criteria: M&S operational requirements and fidelity requirements.

Operational requirements are nonanalytical requirements, in the sense that they do not contribute to resolution of program decisions or their associated MOMs directly. Instead, these requirements define for example,

- hardware and software requirements, e.g., the models and simulations must run on a certain type of workstation under a certain operating system
- pre- and post-processing requirements for M&S data, e.g., M&S inputs or outputs must be converted to special file formats
- operations and training support requirements, e.g., models and simulations cannot have license agreement or operator training requirements because there is no money or no time for training.

Fidelity requirements are the hardest to define. They state how well required M&S functions (or representations, or entities, as well as the interactions between them) must correspond to the real world (see Section 1.2.3) for the M&S results to be acceptable *for the purpose at hand.*[9] This normally requires the development of a notional "error budget," whereby variations in M&S outputs relate to variations in MOM results, which, in turn, correspond to changes in program decisions. Although it is generally possible to specify the *kind* of V&V that needs to be done to support a given level of credibility (for example, face validation versus results validation), the *amount* of V&V required to establish credibility for a particular application will still depend on a clear understanding of how program decisions are affected by M&S outputs.

1.6.1.3 Selecting V&V Tasks

It is now clear why a precise relationship among program objectives, decisions, MOMs, and M&S is essential. The functional, operational, and fidelity requirements developed by the activities previously described constitute a basic checklist of acceptance criteria with which model and simulation characteristics and capabilities can be compared. This comparison is an essential aspect of V&V tailoring, because it justifies objectively the selection of V&V activities. How is this done?

In a typical legacy M&S case, information on model and simulation capabilities is compiled from available documentation, product literature, existing users, and other sources. It is compared to the functional requirements list to determine if any of the required functions are not modeled. In a typical new M&S case, the functional requirements analysis relies heavily on the planning and requirements documentation and on comparison of the conceptual model with the planned uses of the model or simulation. Information on model or simulation operational characteristics, e.g., how much memory it uses, what programming language it is written in, how long it takes to run a typical case, what hardware and operating system is required, what special training and maintenance is required, is obtained. This information is compared to the operational requirements list to determine if additional resources will be required to maintain and operate candidate models or simulations during their application and to decide whether these additional requirements can or should be met. Finally, the fidelity requirements list is compared to the VV&A histories and current results of the candidate models and simulations to determine the applicability of previous V&V and to identify requirements for additional V&V to address the current problem.

Having identified gaps in the V&V state of functional, operational, and fidelity requirements for candidate models and simulations, a VV&A plan can be developed that prioritizes each gap and describes how it will be addressed using the V&V methods most applicable to each model or simulation. Cost and schedule can be estimated for these tasks based on historical data, and risk assessment and mitigation strategies can be developed depending on the way M&S results affect program decisions.[10]

The payoff for giving faithful attention to these aspects of V&V tailoring is that you now have an audit trail of well-defined program objectives and decisions, M&S acceptance criteria, and V&V data that substantiate the use and acceptance of M&S results.

1.6.1.4 Accounting for Uncertainty

Numerous factors make practical application of these guidelines less than straightforward. For example, V&V program costs can be influenced by the requirement for new or specialized training; long-term site visits at national test ranges to support data collection for model and simulation validation; large capital expenditures for hardware and software; unusual technical efforts requiring significant engineering and analysis; and set-up and maintenance of libraries, data bases, threats files, and the like. The most important aspect of uncertainty, however, is the relationship between the level of V&V required to ascertain the credibility of a model or simulation and the process used to develop its software (called the development *paradigm*).

If you're not going to develop a new simulation for use in your program, i.e., you're going to rely on off-the-shelf or legacy models or simulations, you can more or less skip this section. If you'll be building a simulation for use in your program, however, you'll need to modulate the advice given earlier with the practical realities of model and simulation development described in the following paragraphs.

VV&A must parallel model and simulation development to be truly cost-effective. VV&A planning and execution for models and simulations in development cannot occur without two essential ingredients: (a) optimization of the development paradigm by the M&S developer and (b) a thorough knowledge of the total set of program objectives, requirements, and constraints, which are used to tailor the VV&A approach to the needs of the program. (Yet another reason a clear definition of M&S requirements based on program objectives is essential.) Stated another way, models and simulations can be developed several ways; selection of the best development paradigm is based on the unique set of circumstances, constraints, and application particulars defined by your M&S requirements. This development paradigm can influence VV&A requirements heavily. But, if VV&A is involved early enough in the M&S development cycle, it can have a strong influence on the optimization of the development paradigm.

Several M&S development paradigms are available, from the classic single-pass (or waterfall) approach through the recursive (or evolutionary) approaches that include spiral, prototyping, concurrent engineering, and rapid prototyping variations. The more certainty about the detailed requirements of the new simulation, the more likely that it can be generated successfully and economically using a single-pass development paradigm. This approach assumes that the operational expectations and performance requirements of each model or simulation component (and of the model or simulation overall) and the development environment and infrastructure are reasonably well understood and predictable. As more uncertainty is introduced, the need to iterate (or loop) on problem areas increases. In fact, if you can't define your M&S requirements fully early on, iteration becomes an essential strategy to gain sufficient knowledge to justify proceeding to the next phase of development with reasonable confidence, effectively controlling and managing risk. The complexity of scheduling and budgeting model or simulation development and VV&A activities likewise increases with uncertainty, so that contingency allowances (often called the *management reserve*) must be factored into cost estimates, schedules, and program plans.

Perhaps the simplest way to visualize selection of an appropriate model or simulation development paradigm is to imagine a continuum of certainty and uncertainty. Figure 1-2 lists several of the key attributes that help an M&S developer figure out where the development program belongs along this continuum. The left side of the figure depicts a high degree of certainty about key development factors. This means that the decision makers (you) and the M&S developers have a secure knowledge of and confidence in the technologies, systems and components, similar M&S configurations, communications, protocols, data and data bases, operational requirements, scenarios, and other important data needed to define the model or simulation fully. As long as the requirements are stable and predictable, the waterfall model works well. As the uncertainty about key M&S development factors increases, however, the developer is driven toward iterative development paradigms.

A rapid prototyping paradigm is best used when M&S requirements cannot be defined completely at the beginning of the program. In this approach, part of the model or simulation is built and tested, exercised, or demonstrated to enable the users to work with it and thus help define the next, expanded set of requirements. The process repeats until the user (you) is finally satisfied that the product does all of the essential things. Rapid prototyping is highly adaptive and can be used at will almost any time that a high-risk or unknown part of the model or simulation must be expanded. By building an executable piece of the model that can be demonstrated to the customer and user community (e.g., your M&S shop), feedback and refinement can occur very efficiently. Rapid prototyping is extremely useful in developing and evaluating requirements, proving early design concepts, demonstrating the graphical user interfaces and human interactions, proving critical algorithms, and evaluating the environment and infrastructure. It can be inserted anywhere in the development cycle to help solve technical problems and can be used with virtually any of the other development paradigms.

High Degree of Certainty High Degree of Uncertainty

• Known technologies		• Unproven technologies
• Known, stable requirements		• Unstable requirements
• Reused, VV&A'ed parts		• Mostly new, untried parts
• Stable design		• Fluid design
• Known communication network		• Undecided communication network
• Predictable performance		• Unknown performance
• Strong tool base		• Sporadic tool application
• Certified data sources		• Indefinite data sources
• Known operational objectives		• Vague operational objectives
• Trained participants		• Nondedicated participants
Waterfall Model	< < <---------------> > >	***Recursive Models***
Minimum V&V	< < <---------------> > >	***Maximum V&V***

Figure 1-2. The Certainty-Uncertainty Continuum

But what's all this got to do with the scope of VV&A activities? Simply put, VV&A activities strongly depend on the development paradigm. Generally, the more uncertainty in M&S requirements, the more effort will be expended on VV&A. It is here that a list of VV&A activities, those that normally would be completed in a comprehensive effort, can be of great help. Because such a VV&A list defines a very rich set of activities, only higher level VV&A efforts will attempt them all. A moderate V&V approach, on the other hand, reduces both the intensity and the number of specific activities planned, focusing on those that are most important to the success of the M&S development program as defined by program requirements. Minimum efforts focus sharply on essential activities.

1.6.1.5 Scoping and Cost Conclusions

It's clear that whether you're using legacy models and simulations or building new ones, defining your M&S requirements based on your specific application is essential to the cost-effectiveness of any VV&A efforts. If you can't (or won't) spend the money to define those requirements, chances are you're going to waste a good portion of whatever V&V dollars you do spend.

VV&A planning should not become a contest to provide the absolute lowest cost effort nor, at the other extreme, to provide more elaborate procedures and analyses than are required. Cost-effective VV&A seeks the best value balance between program needs and real-world constraints. When faced with budgets that appear too low to accomplish the VV&A activities suggested by program requirements for M&S credibility, trade-offs have to be made. These trade-offs should prioritize those activities that have the greatest return on investment (ROI) and that instill and confirm the greatest degree of confidence in the model or simulation. Thus, final selection of VV&A activities must be driven by program particulars: discrete requirements, defined needs, known problem areas, high-risk and critical items, and availability of tools, methods, and key staff.

Tailoring VV&A activities requires careful analysis of M&S requirements, an understanding of the development paradigm (when new models and simulations are being developed), knowledge of problem areas and relevant technologies, knowledge of and access to authoritative data sources, and understanding of the M&S environment and infrastructure. The amount of uncertainty governs the amount of VV&A; that's just common sense. When applied in good faith, as opposed to a desire to check a box, VV&A can add substantial value to the integration of M&S into your program, and its cost can be completely justified by validating the conceptual model, reducing rework, detecting problems early, stabilizing the M&S suite chosen for use, improving analytical efficiency, correlating results, ensuring compatibility, and supporting test and evaluation.

1.6.2 Key Players, Roles, and Functions

This section describes the personnel needed to perform VV&A, the roles and responsibilities of major players, the need for independence in V&V, and trade-offs between independence and ignorance. We explore the appropriate roles of M&S sponsors, developers, V&V agents, accreditation agents, and accreditation authorities. Before we discuss the substance of this section, however, we have to get a few definitions out the way. According to DoDI 5000.61, the following are the accepted definitions of the terms used in the first sentence of this paragraph:

- *M&S Application Sponsor*—The organization that utilizes the results or products from a specific application of a model or simulation
- *Accreditation Agent*—The organization designated by the application sponsor to conduct an accreditation assessment of an M&S application
- *M&S Developer*—The organization responsible for managing or overseeing models and simulations developed by a DoD Component, contractor, or Federally Funded Research and Development Center[11]
- *Validation (or Verification) Agent*—The organization designated by the M&S application sponsor to perform validation (or verification) of a model, simulation, or federation of models and/or simulations.

In a typical scenario, the application sponsor (the one who needs M&S to solve a problem or answer a question) will designate an accreditation agent, who is responsible for organizing, coordinating, and executing a comprehensive VV&A program that will guarantee the credibility of model and simulation results when used for the sponsor's application. The accreditation agent may further designate a V&V agent who will be responsible for producing the V&V data used to accredit the model, or the agent may act as his or her own V&V agent. The M&S Developer is typically designated by the application sponsor to oversee M&S development activities and to ensure coordination with the V&V agent, but the application sponsor also may retain the duties of M&S Developer. In any case, the exact relationship between these organizational entities can have a bearing on the credibility of the outcome of VV&A activities.

A common (mis)perception holds that V&V must be conducted completely independent of the M&S developer, lest the results be tainted by the demands of advocacy, whence the *I* in IV&V. The M&S developer, however, is (and should be) an essential and integral part of V&V, contributing greatly to its efficiency because the developer is intimately familiar with the design and code details and has been involved in the intricacies of development from the start. The developer understands (or should understand) the requirements best and in the best of cases has maintained close contact with the application sponsor. It is also true, however, that the developer has a vested interest in making the product look good. The need for some kind of independent assessment of the developer's product seems like a common sense risk reduction strategy.

But there is a down side to independence. Totally independent V&V efforts by the V&V agent can retrace much of the work already done by the M&S developer. Rework is fine if the developer is trusted to provide much of the essential information for the V&V agent. Sometimes, however, the relationship between the V&V agent and the developer can become adversarial, with the V&V agent taking on functions of a Government Inspector General. This opposition can burden the development process with unnecessary baggage that you will ultimately have to pay to carry, all in the name of independence. The question that must be answered is, "How much independence can I afford?"

No real hard and fast rules dictate how much *I* to put in V&V. Some notional V&V roles and responsibilities that worked in the past are shown in Table 1-1, but the final decision must be derived from the trade-off between the M&S budget and the level of confidence and trust that can be placed in the M&S developer. Don't forget that V&V also may be performed in-house by the application sponsor. Frequently, the M&S developer performs the verification of the model or simulation with V&V agent oversight and assists the V&V agent or the application sponsor during validation.

Table 1-1. Typical VV&A Responsibilities

Activity	*Party*			
	V&V Agent	M&S Developer	Application Sponsor	Accreditation Agent
V&V Acceptability Criteria Report	Assists		Responsible	Assists
Accreditation Plan			Responsible	Performs
V&V Plan	Responsible, Performs	Assists	Uses	Uses
Verification	Responsible	Assists		
Validation	Responsible	Assists		
V&V Report	Responsible, Performs	Assists	Uses	Uses
Acceptability Assessment Report	Assists			Responsible
Accreditation	Assists		Responsible, Performs	Assists
Accreditation Report	Assists		Responsible	Performs

Table 1-1 uses the terms: *Responsible*, *Performs*, *Assists*, and *Uses*. *Responsible* means that the listed party ensures that the specified activity is accomplished. *Performs* means that the listed party carries out the technical work associated with the listed activity. *Assists* means that the listed party helps the responsible or performing party with the activity. *Uses* means the listed party employs the product of the listed activity in performance of some function listed later in the table. Remember that Table 1-1 is only a suggested list of interactions and responsibilities. Ultimately, you must decide how much independence is necessary and affordable.

1.6.3 The Importance of Configuration Management

This section describes the relationship between sound configuration management and cost-effectiveness of VV&A. It provides guidelines for evaluating or implementing configuration management procedures.

Software Configuration Management (C/M) is a development life-cycle process through which the integrity and continuity of software development, upgrades, and maintenance are recorded, communicated, and controlled. C/M can have a profound impact on the sustainability of M&S credibility you have worked so hard to attain through V&V. The key to maintaining the shelf life of V&V work is a structured, workable, and well-maintained C/M process that is integrated with model and simulation development. Because the magnitude of the C/M problem will vary depending on the use of legacy models or simulations, only the most general comments will be given here.[12] It is not an overgeneralization to state, however, that V&V not integrated with C/M will result in repetitive efforts and wasted resources.

But what are the elements of a good C/M process? In general, four major characteristics are the hallmarks of sound C/M practice:[13] (a) a well-defined baseline; (b) standard baseline test cases and data sets; (c) well-defined, coordinated, and supported testing program; and (d) current, thorough documentation. Whether you are using legacy models and simulations or developing your own, you should evaluate the C/M process for these characteristics. If it has all four, you can be reasonably sure that the model or simulation is well-managed and controlled. Some special considerations apply to legacy and new models and simulations, and these are discussed in Chapter 3.

1.6.4 Credibility of M&S Data

All M&S are driven by data, either as direct inputs by the user or as embedded constants that drive simulation characteristics. As perfect as the equations, algorithms, and software design of a M&S may be after conceptual model verification and validation and design verification, it will probably fail results validation if the data that drive the simulation are inaccurate or inappropriate for the task at hand. Data credibility is a major driver of M&S credibility.

But how can the credibility of M&S data be quantified? And what standards should be proposed so that the credibility of data used by multiple users is uniform? Data standards benefit all M&S users by providing increased data credibility, reduced need for data translation, interoperability with the operational community, and M&S reuse. Without data standards, interoperability between models and simulations is much more difficult to achieve. Data definitions common to different systems are needed, definitions that are formal and consistent and that use data standardization policies, procedures, and methodologies.

The M&S community has been wrestling with this issue for some time now, and a final pronouncement of standards, procedures, and guidelines for the certification of data credibility has not been made. Several key concepts have emerged, however. Data verification and validation definitions, processes, and procedures that parallel the M&S definitions, processes, and procedures have emerged. For data, the decision is called *certification* as opposed to *accreditation*. Hence the term data VV&C instead of VV&A. Additionally, data VV&C is viewed from two different perspectives: that of the data producer and that of the data user. The key definitions are as follows:

- Data verification establishes that the data produced conform to the specification.
- Data validation establishes that the data accurately represent the real world.
- Certification establishes that the data are suitable for a specific use.

Currently, each Service tracks data about their models and simulations and stores them at the service's own required level of detail in service-specific format. DMSO has been coordinating an effort to standardize the level of detail, format, and accessibility for all DoD M&S data, including VV&A and data VV&C information. These data will be centrally controlled but accessed in a distributed environment. They will provide information critical to M&S planners and the basis for model and simulation life-cycle management. Additional discussion of data VV&C is in Chapter 3.

1.7 Roadmap to This Guide

Now that you've had an overview of VV&A, you're ready for more detail. Chapter 2 provides a set of governing principles of VV&A based on the experience of Government, industry, and academic experts. Chapter 3 introduces the basics of VV&A processes and sets the context for Chapter 4, which deals with the details of VV&A techniques. Chapter 5 discusses combining V&V information into a sound accreditation decision, and Chapter 6 completes this guide by discussing common reporting formats that will simplify the maintenance of an audit trail of V&V and accreditation support activities.

Endnotes

---

[1] *V&V* is the term applied to the technical work that supports a decision ("accreditation") to use a model or simulation. The terms are defined later in this section.

[2] For example, a flight training simulator does not need to kill the pilot if he or she crashes the plane during training to accomplish the goal of teaching the pilot not to crash.

[3] Inclusion of the model or simulation in one of the Information Analysis Center (IAC) model repositories can be a good indication of community acceptance of M&S results. IACs are run by the Defense Logistics Agency (DLA) to support a wide variety of DoD analysis needs. DMSO's Modeling and Simulation Resource Repository (MSRR) is, likewise, a good source for M&S resources that are considered authoritative.

[4] *V&V'ed*, in the vernacular, although some object to this casual use of technical terms.

[5] For example, loss of at least one fly-by-wire aircraft has been attributed to M&S inadequacies. See *The Day the Phones Stopped* by L. Lee (Donald I. Fine, Inc., 1991).

[6] See Army Regulation (AR) 5-11 and the associated DA PAM 5-11; Air Force Instruction 16-1001; draft SECNAVINST 5200.1; DoDD 5000.59; and DoDI 5000.61.

[7] VV&A efforts conducted under Military Operations Research Society (MORS), Distributed Interactive Simulation (DIS), and Susceptibility Model Assessment with Range Test (SMART) auspices are notable in this regard.

[8] *MOM* is a generic term that encompasses all measures of value, including Measures of Performance (MOPs), Effectiveness (MOEs), and Outcome (MOOs).

[9] This does not imply an absolute standard of fidelity for all applications but rather a level of fidelity considered good enough. The good must not become the enemy of the best.

[10] Another reason to spend some time defining your program objectives, decisions, and M&S requirements.

[11] For the purposes of this discussion, we will include the organization (government or contractor) responsible for actually building the software under the term *developer*.

[12] See, however, *Configuration Management Requirements Study*, available from the JTCG/AS (JTCG/AS-95-M-005), which discusses DoD and MIL-STDs for software C/M.

[13] Comments on C/M for legacy models and simulations are taken from the study cited in Footnote 12. The goal of the study was to identify common requirements for the C/M of legacy models and simulations, to compare these requirements to current practice, and to make recommendations for improvement.

```
                    b                         d
                    b                         d
                    b                         d
n nnn     oooo      b bbb     oooo       ddd d   y    y
nn   n   o    o     bb   b   o    o     d   dd   y    y
n    n   o    o     b    b   o    o     d    d   y    y
n    n   o    o     b    b   o    o     d    d   y    y
n    n   o    o     bb   b   o    o     d   dd   y   yy
n    n    oooo      b bbb     oooo       ddd d    yyy y
                                                      y
                                                 y    y
                                                  yyyy


                              ll                                     00000
                               l     t                              0     0
                               l     t                              0    00
 cccc    w     w   aaaa        l   ttttt    eeee              aaaa  0   0 0
c    c   w  w  w       a       l     t     e    e                 a 0  0  0
c        w  w  w   aaaaa       l     t     eeeeee             aaaaa 0 0   0
c        w  w  w  a    a       l     t     e                 a    a 00    0
c    c   w  w  w  a   aa       l     t  t  e    e     ..     a   aa 0     0
 cccc     ww ww    aaaa a     lll     tt    eeee      ..      aaaa a 00000


                       i             t
                                     t
p ppp    r rrr        ii    n nnn  ttttt   eeee   r rrr    ssss
pp   p   rr   r        i    nn   n   t    e    e  rr   r  s    s
p    p   r             i    n    n   t    eeeeee  r        ss
p    p   r             i    n    n   t    e       r          ss
pp   p   r             i    n    n   t  t e    e  r       s    s
p ppp    r            iii   n    n    tt   eeee   r        ssss
p
p
p
```

Job: cwalte.a07953
Date: Mon Oct 13 12:26:23 1997

Chapter 2 — Principles

Chapter 1 reviewed Verification, Validation, and Accreditation (VV&A) terminology and introduced some special topics that are of management-level concern with respect to VV&A efforts. This chapter delves a little deeper into the arcana of VV&A and discusses some principles that should be of concern to those tasked to develop and implement a VV&A program. The following principles are by no means the last word in VV&A orthodoxy, but they do distill the corporate memory of VV&A experts from government, industry, and academia into a convenient synopsis of hints that should help both the VV&A novice and the veteran avoid common pitfalls.

In the following paragraphs, 12 guiding principles of VV&A are proposed for your consideration and use. Some of these principles are no more than applied common sense; others require the nuance gained by experience to be appreciated. These principles form the basis for the whole concept of VV&A and for 76 recommended V&V techniques (described in Chapter 4) that can be used throughout the modeling and simulation (M&S) life cycle. Understanding and applying these principles is essential not only to the success of an M&S development effort, but also to the efficient application of VV&A resources and the credible integration of M&S into your specific application.

2.1 Principle 1

There is no such thing as an absolutely valid model.

By definition, a model is an *abstraction* or *approximate* representation of something. No model is ever totally representative (Banks *et al.*, 1996, p. 407) and no model ever can be absolutely correct (Shannon, 1975).

As depicted in Figure 2-1 (Shannon, 1975; Sargent, 1992), an increase in model credibility infers a similar increase in model development costs. At the same time, the model utility also will increase but most likely at a decreasing rate. The point of intersection of these two curves is different for each M&S application.

Even at the extremes, however, the word *absolutely* must be interpreted in light of the application in which the model or simulation is to be used. A model can be *absolutely* invalid for one application but reasonable (within limits) for another. Thus, the same set of V&V data can be used to accredit a model for one application and yet point to the need for more development or additional V&V for another application.

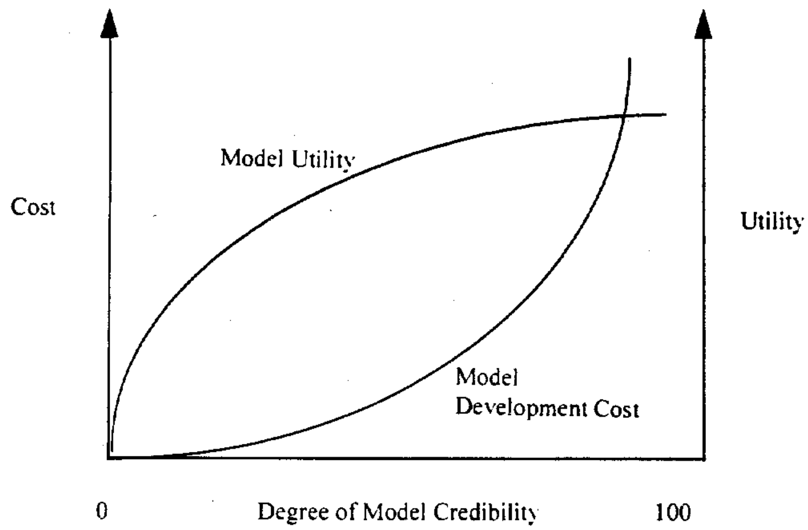

Figure 2-1. Model Credibility versus Cost and Utility

The goal of V&V is to exercise the model using a range of inputs, to identify and diagnose anomalous results, and to fix any problems encountered. Complete V&V would require the model or simulation to be tested and examined under all possible combinations of input conditions. Such combinations of input conditions easily can generate millions of logical execution paths that need to be checked for validity. Although some automated tools are available to assist in this task, time and budget constraints usually preclude exhaustive testing. As a result, the scope of V&V is usually tailored to the credibility requirements of the application. How much V&V is required depends on the intended use of the model or simulation. The greater the extent of the intended use, the more V&V will be required. It is impossible, however, to test all facets of a complex simulation.

2.2 Principle 2

VV&A should be an integral part of the entire M&S life cycle.

Despite the increased awareness of and attention to VV&A in recent years, many in the M&S community still refer to VV&A as if it were a single phase or step in the M&S development life cycle. The verb *to VV&A* is common among M&S practitioners, as are phrases such as "the model was VV&A'ed." Although M&S practitioners may know what they mean by these terms, their usage devoid of context has given many the incorrect impression that VV&A is a single task, rather than a process.

In reality, VV&A is a continuous activity performed throughout the entire life cycle of an M&S application, as shown in Chapter 3 (Figure 3-7). The sequential representation of the arrows in this figure is intended to show the general direction of M&S development, but the life cycle should not be interpreted as being strictly sequential. It is iterative in nature, and reverse transitions between phases of the life cycle are typical. Every phase of the life cycle has an appropriate set of VV&A activities whose breadth and depth is, in part, determined by the specifics of the application for which the simulation is being accredited. Deficiencies identified by VV&A may necessitate a return to an earlier development stage for revision and improvement.

For new M&S developments, conducting V&V for the first time in the life cycle only after the simulation has been developed completely is analogous to the teacher who gives only a final examination (Hetzel, 1984). No opportunity is provided throughout the semester to notify the student of serious deficiencies. Severe problems may go undetected until it is too late to do anything but fail the student. Frequent tests and homework throughout the semester are intended to inform students about their deficiencies so that they can take corrective action to improve their knowledge and performance as the course progresses.

The situation with VV&A is analogous. VV&A activities conducted throughout the M&S life cycle are intended to reveal deficiencies that might be present as development progresses from problem definition to the analysis of results. Tying VV&A to the M&S life cycle offers the opportunity to detect errors as early as possible and correct them at less cost and risk to the overall program. Too often there is a rush to implement a simulation, or "write the code." Sometimes simulations are built without formal specification of the requirements, resulting in a simulation that may not be relevant to the original problem to be answered by the use of this simulation. As a result, VV&A of the simulation becomes the only credibility assessment tool. And detecting and correcting the major modeling errors that VV&A will uncover at this stage is very time-consuming, complex, and expensive (Nance, 1994). *Correction of errors early in development always costs less than correction of errors later. If you are worried about the cost of VV&A, it is better to spend a little up front than a lot later.*

For legacy models and simulations, the situation appears a bit different on the surface, but in reality the same principle applies. Recall that by definition, legacy models and simulations were built before the development and implementation of consistent standards for VV&A. In practice, this has meant that little or no formal VV&A has been conducted for legacy models and simulations at any stage in their development life cycle. For this (and other) reasons, it has become common practice to use the term *legacy* as a pejorative modifier to "models and simulations." Recent postdevelopment VV&A experience with several widely used legacy models and simulations indicates that this practice is unjustified, however. For example, in the Susceptibility Model Assessment with Range Test (SMART) Project, several legacy models and simulations used by all Services to support acquisition and testing decisions were put through a rigorous V&V program with very positive results. This favorable reassessment is the result of efforts (a) by expert users to work out many of the problems associated with good legacy models and simulations over many years of use, (b) by frequent user group meetings, and (c) by reasonably well-disciplined configuration management practices developed through years of experience. The fact that such models and simulations fared well when subjected to the detailed scrutiny of formal VV&A performed after development should come as no surprise. In fact, the long period of use, modification, and consensus-building among the user community constitutes a form of face validation. (See Chapter 4 for a discussion of face validation as a V&V technique and its limitations.) Users, developers, and sponsors of the most frequently used legacy models and simulations have come to the realization in recent years that VV&A needs to be integral with development, and many have begun to incorporate formal VV&A into their development plans and budgets. Consequently, even for legacy models and simulations, it is becoming *de rigeur* for VV&A to be an integral part of the development life cycle.

2.3 Principle 3

A well-formulated problem is essential to the acceptability and accreditation of M&S results.

It has been said that a problem correctly formulated is half-solved (Watson, 1976). Albert Einstein once indicated that the correct formulation of a problem was even more crucial than its solution. The accuracy of the formulated problem greatly affects the accreditation and acceptability of M&S results. Insufficient problem definition and inadequate sponsor involvement in defining the problem are two of the most significant problems in the development and use of M&S. It must be recognized that, if problem formulation is poorly conducted, resulting in an incorrect problem definition, no matter how fantastically the problem is solved, the M&S results will be irrelevant. Balci and Nance (1985) present an approach to problem formulation and propose 38 indicators to use in assessing the formulated problem.

2.4 Principle 4

Credibility can be claimed only for the intended use of the model or simulation and for the prescribed conditions under which it has been tested.

Many people think that once simulation results have been compared with real-world data or test data, the simulation is "validated" for all time for all applications. Nothing could be further from the truth. The definitions for both validation and accreditation clearly require assessment of a model or simulation against a specific intended use. Global VV&A simply does not exist.

As stated in Principle 3, it is vitally important to the VV&A effort to define the intended use of the model or simulation clearly at the beginning of its application. (See Chapter 3 for an additional discussion of problem definition.) A well-defined problem statement will provide the necessary framework from which credibility of the model or simulation can be established and assessed.

Application objectives and requirements dictate how faithful the representation of a process, phenomenon, or system must be when compared with the real world for simulation results to be considered useful. Sometimes, 60 percent representation accuracy may be sufficient; sometimes, 95 percent accuracy may be required, depending on the type or importance of decisions that will be made based on the simulation results. Because the requirement for accuracy varies with the intended application, it is clear that a model's or simulation's credibility must be judged with respect to application-specific requirements and objectives. The adjective should always be used in front of terms such as credibility, validity, or accuracy to indicate that the judgment of validity has been made with respect to application-specific requirements.

Many factors can influence the intended use of a model or simulation. For example, modifications to simulations over time can introduce changes that require the results to be compared again with test data. In some cases, the simulation may have changed so much that the original validation data set has been rendered obsolete or inapplicable. Even if the data set is still acceptable, however, the simulation is still only valid over the range of conditions under which it is compared with test data. If the range of conditions is small, so is the range of validity.

Another factor that can influence the intended use of a model or simulation is the fact that the relationship between simulation inputs and outputs can be affected by the characteristics of the input conditions. The relationship that applies to one set of input conditions may produce absurd results when another set of input conditions is used. An intuitive example may help to illustrate this point. A model of a traffic intersection can be developed assuming a constant arrival rate of vehicles. (This is a reasonably valid assumption during the evening rush hour, for example.) When compared with actual traffic data at rush hour, the credibility of the model may be judged to be adequate with respect to evening rush hour conditions. The simulation will show erratic or invalid behavior when run under non-rush hour conditions, however. During this period, the arrival rate of vehicles is not constant, and a different simulation approach is required. Thus, the simulation could not be used to predict the behavior of a traffic intersection under non-rush hour conditions, even though it had been "validated" for rush hour conditions.

2.5 Principle 5

M&S validation does not guarantee the credibility and acceptability of analytical results derived from the use of simulation.

Model or simulation validity is a necessary, but not a sufficient, condition for the credibility and acceptability of the analytical results derived from use. Validity is judged with respect to the M&S objectives and requirements as they are defined. If the M&S objectives and requirements are incorrectly identified or the problem is improperly formulated, analytical results derived from the use of a model or simulation can be invalid or irrelevant; however, the model or simulation can still be found to be sufficiently valid by comparing it with the improperly defined system and requirements and with respect to the incorrectly identified objectives.

2.6 Principle 6

V&V of each submodel or federate does not imply overall simulation or federation credibility and vice versa.

Because a model is an abstraction of a system, with inherent assumptions, limitations, and approximations, it is unreasonable to expect perfect representation of all aspects of the modeled system. The credibility of each submodel or federate is judged to be sufficient with some error that is acceptable with respect to the M&S application objectives. Each submodel may be found to be sufficiently credible, but this does not imply that the whole M&S application is sufficiently credible. The allowable errors for the submodels may accumulate and become unacceptable for the whole model or simulation. Therefore, the whole model or simulation must be tested even if each submodel has been tested individually and found to be sufficiently credible. This same requirement applies to federations as it does to individual models and simulations.

Similarly, the determination that a federation has sufficient credibility cannot and does not imply that the individual federates are credible. Unfortunately, assumptions of federate credibility often are made to expedite the VV&A of a large federation. It must be remembered that submodel or federate V&V often rests on a previous or partial V&V effort that may not be relevant to the use of the model in the new application. The assumption that a model has undergone some level of scrutiny in the form of VV&A, when that VV&A actually has not been performed, leaves a significant gap in the understanding of the model and the resulting credibility assessment. When such assumptions are made on the VV&A of federate models, the investment of time and money on the VV&A of the federation results in a credibility assessment with little or no basis in reality and of limited use to the accrediting authority. Both the federation and its underlying federates and submodels must be verified and validated for the federation to be accredited for a specific use.

2.7 Principle 7

Accreditation is not a binary choice.

V&V results are not like the answers to Twenty Questions; very rarely are they simply "Yes, the model is good" or "No, the model is bad." Because a model is an abstraction of a system, with inherent assumptions, limitations, and approximations, it is unreasonable to expect perfect representation of all aspects of the modeled system when compared with test or other data. Consequently, it is more useful to consider the outcome of V&V activities in terms of a degree of confidence in M&S results that is expressed on a scale from 0 to 100, where 0 represents absolutely incorrect and 100 represents absolutely correct.

Figure 3-7 in Chapter 3 illustrates a range of responses available to the accrediting authority, including accredit the model, accredit the model with limitations to its use, modify the model, refer for additional V&V, or don't accredit the model. Accreditation agents, in particular, need to assist the accrediting authority in understanding his choices. Accreditation is not a fait accompli, as in the quick answer of "well, when it's accredited," which fails to acknowledge that the model may not be accredited. Neither is accreditation a yes or no decision. Chapter 5 provides additional amplification regarding the accreditation decision and is a must read for both accrediting authorities and program managers.

2.8 Principle 8

VV&A is both an art and a science, requiring creativity and insight.

Many people believe that because VV&A techniques are well-defined at the technical level, the integration of these techniques to establish model or simulation credibility for particular applications at least is similarly straightforward. This is not true. Cost-effective VV&A requires creativity and insight. It is not a checkmark in the box.

One must understand thoroughly the whole M&S application to design and implement effective tests and identify adequate test cases. Knowledge of the problem domain, expertise in the M&S methodology, and prior modeling and V&V experience are required for successful VV&A. It is not possible, however, for one person to understand fully all aspects of a large and complex model, especially if the model is stochastic and contains hundreds of concurrent activities.

The model or simulation developers are usually the most qualified to show the creativity and insight required for successful V&V because they are intimately knowledgeable about the model or simulation. They usually are biased, however, when it comes to testing their own models and simulations and, therefore, cannot be solely responsible for V&V. This limitation increases the difficulty inherent in V&V.

False beliefs exist, as indicated by Hetzel (1984) in referring to testing, beliefs that are often associated, however wrongly, with VV&A: testing is easy; anyone can do testing; no training or prior experience is required. As with testing, the difficulty of model or simulation V&V must not be underestimated. V&V must be well planned. It must be administered by the proponents and agents who are responsible for the model or simulation application. If V&V is delegated to contractors, oversight is required by the V&V agent. VV&A must also involve subject-matter experts to retain the focus on the needs of the warfighter. VV&A is a team effort.

2.9 Principle 9

The success of any VV&A effort is directly affected by the analyst.

The impact of the analysts who participate in the model or simulation application and, more particularly, in the VV&A of the model or simulation, is often overlooked. Analysts are key players in the use of M&S, from assisting in defining the problem to selecting the model or simulation to be used to running the simulation to interpreting the results. Military analysts frequently are called upon to provide the analytical perspective but also may serve as subject matter experts for applications involving warfare operations from their personal fields of expertise. Analysts can be found in virtually all roles within the M&S life cycle, including those of V&V agent, program manager, M&S proponent, and even accreditation authority.

Traditional training for analysts has not always included sufficient instruction in the use of M&S or the development of the technical skills required for VV&A. Part of the VV&A process itself, therefore, must consider the qualifications of the VV&A team, most notably, the analysts who are involved extensively in the M&S process. Emphasis must be placed on carefully achieving the right balance of education, experience, practical knowledge, and technical skills. The tendency to assume the qualifications of an analyst based on experience that is no longer timely or on an advanced degree should be avoided.

2.10 Principle 10

VV&A must be planned and documented.

Principle 2 stated that VV&A is not a phase or step in the M&S life cycle—it is a continuous activity. Ad hoc or haphazard V&V does not provide a reasonable measurement of model accuracy. Hetzel (1984) points out that "such testing may even be harmful in leading us to a false sense of security." Careful planning is required for successful VV&A efforts. Tests should be identified, test data or cases should be prepared, tests should be scheduled, and the whole VV&A process should be documented.

2.11 Principle 11

V&V requires some level of independence to minimize the effects of developer bias.

Many people have heard the term IV&V, where the "I" stands for independent. Some in the M&S community insist that V&V must be as independent as possible from the developer of M&S software to minimize the effects of bias: the fox must not be allowed to guard the chickens. It is certainly true that simulation testing and evaluation is more meaningful when conducted by someone who has no vested interest in the outcome. Model developers, who have the most knowledge of the model, may be the least unbiased in testing and evaluating their own products. This is understandable. Beyond the natural resistance to criticism, they may fear the repercussions that negative results may have on their performance appraisal, the credibility of their organization, and the prospect of future contracts. It is also true, however, that an insistence on excessive independence can lead to duplication of effort between the IV&V agent and the developer, to needlessly adversarial relationships, and to increased costs. It is possible, after all, to mistake ignorance for perspective. Neither extreme of independence is worth pursuing seriously, but no real hard and fast rules exist for how much "I" to put in IV&V between these extremes. The final decision must be derived ultimately from the trade-off between your budget and your level of confidence and trust in the model or simulation developer, as well as the requirements of your management chain to demonstrate independence.

2.12 Principle 12

Successful VV&A requires data that have been verified, validated, and certified.

The credibility of M&S results is related directly to the credibility of data used as input to or resulting from model use. Data need to be reviewed for accuracy and consistency, as described in Section 1.6.4. Guidelines are being developed to provide insight into the tools, techniques, processes, and procedures that assist the model user in determining data credibility.

References

Balci, O., & Nance, R.E. (1985). Formulated problem verification as an explicit requirement of model credibility. Simulation, 45(2), 76–86.

Banks, J., Carson, J.S., & Nelson, B.L. (1996). Discrete-event system simulation (2nd ed.). Englewood Cliffs, NJ: Prentice-Hall.

Hetzel, W. (1984). The complete guide to software testing. Wellesley, MA: QED Information Sciences.

Nance, R.E. (1994). The conical methodology and the evolution of simulation model development. Annals of Operations Research, 53, 1–46.

Sargent, R.G. (1992). Validation and verification of simulation models. In J.J. Swain, D. Goldsman, R.C. Crain, and J.R. Wilson (Eds.), Proceedings of the 1992 Winter Simulation Conference (pp. 104–114), Arlington, VA, 13–16 December 1992. Piscataway, NJ: IEEE.

Shannon, R.E. (1975). Systems simulation: The art and science. Englewood Cliffs, NJ: Prentice-Hall.

Watson, C.E. (1976). The problems of problem solving. Business Horizons, 19(4), 88–94.

Additional Reading

Balci, O., & Sargent, R.G. (1981). A methodology for cost-risk analysis in the statistical validation of simulation models. Communications of the ACM, 24(4), 190–197.

Balci, O., & Sargent, R.G. (1984). Validation of simulation models via simultaneous confidence intervals. American Journal of Mathematical and Management Sciences, 4(3&4), 375–406.

Johnson, M.E., & Mollaghasemi, M. (1994). Simulation input data modeling. Annals of Operations Research, 53, 47–75.

Law, A.M., & Kelton, W.D. (1991). Simulation modeling and analysis (2nd ed.). New York, NY: McGraw-Hill.

Schlesinger, S., et al. (1979). Terminology for model credibility. Simulation, 32(3), 103–104.

Chapter 3 — Processes

3.1 Background and Overview

The preceding chapters provide the foundations upon which an understanding of Verification, Validation, and Accreditation (VV&A) can be built. The underlying philosophy and the guiding principles associated with VV&A serve as navigational aids in the process of VV&A application and implementation. Chapter 3 builds on this foundation, describing for the use of the VV&A practitioner the fundamental elements associated with a generic VV&A process.

Because the VV&A process shares a symbiotic relationship with the M&S life cycle development process, introductory sections focus on the development process as well as on some of the more commonly used development paradigms. These sections are followed by a description of the generic VV&A process and the application of this process to the High-Level Architecture (HLA) federation development process. Concluding sections discuss VV&A processes as defined by some of the major DoD M&S communities, including those employing legacy simulations, those developing new models and simulations, and those associated with Distributed Interactive Simulation (DIS) or Aggregate-Level Simulation Protocol (ALSP) applications. The relationship of these processes to the generic VV&A process is then explored.

Using the Defense Science Board's definition that "anything short of warfare is a simulation," the spectrum of M&S to be addressed by this document is quite broad and can be represented best as a three-dimensional cube composed of M&S classes, M&S functional areas, and M&S implementations (Figure 3-1).

The dimensions of the M&S cube are defined in the following paragraphs.

3.1.1 M&S Classes

All classes of M&S involve computer programs that either replicate military systems or support actual use or testing of military systems. Some M&S involve hardware, actual military equipment, or personnel. Specific classes include the following:

- Constructive—computer simulations, including man-in-the-loop and hardware-in-the-loop M&S
- Virtual—weapon system simulator forces
- Live—instrumented tests and exercises.

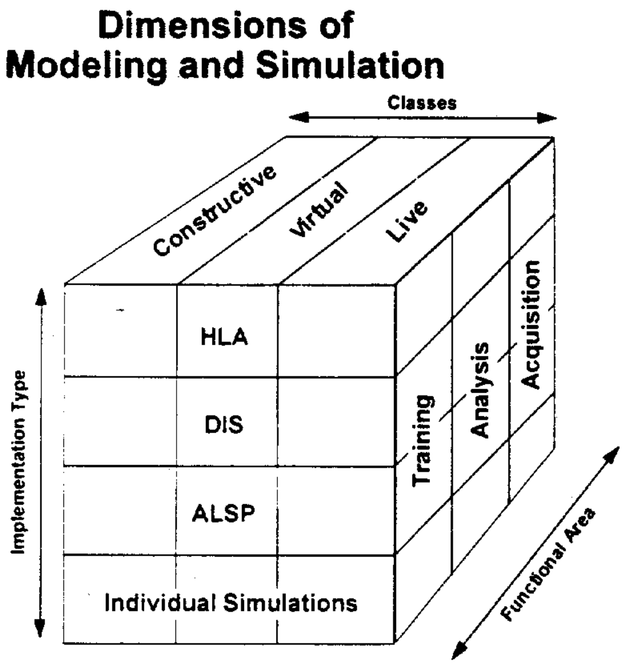

Figure 3-1. The M&S Cube

3.1.2 M&S Functional Areas

As defined by the DoD Modeling and Simulation Master Plan, there are three functional areas:

- Acquisition
- Analysis
- Training

3.1.3 M&S Implementation Types

M&S may be implemented using models and simulations that either stand alone or are brought together in some form of federation. Federations of M&S may be in one place or may be distributed geographically or across multiple platforms. The networking of a computer simulation with a stimulated piece of hardware may be considered a federation, even though the two elements are sitting side-by-side. Alternatively, a federation might involve live players interacting across continents with computer simulations, both constructive and virtual. Current methods of federating M&S include ALSP, DIS, and the HLA, which is designed to provide a common technical framework that promotes and supports interoperability and reuse of M&S across DoD.

3.2 Definitions

The definitions for the terms verification, validation, and accreditation, provided in Chapter 1, are repeated here to set the stage for the following discussions of the VV&A process. These definitions reflect the DoD position on VV&A as defined in DoD Directive 5000.59, M&S Management, and DoD Instruction 5000.61, M&S VV&A.

- Verification—The process of determining that a model implementation accurately represents the developer's conceptual description and specifications
- Validation—The process of determining the manner and degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model
- Accreditation—The official certification that a model or simulation is acceptable for use for a specific purpose

3.3 The M&S Life Cycle

3.3.1 Process Description

The M&S life cycle underlies all supporting processes such as VV&A, testing, configuration management, quality assurance, and data development. Figure 3-2 depicts the M&S life cycle.

The life cycle is initiated by the definition of a problem that a given user, or application sponsor, needs resolved. Associated with the problem definition is a set of high-level requirements encapsulating the user's objectives. This stage cannot be overemphasized, because all too often, M&S is used without a clear definition of the problem to be solved or the questions to be addressed.

Once the preliminary requirements have been defined, a course of action is selected. The user determines if modeling and simulation is the best approach to obtaining the desired solution. It should be noted that M&S is only one tool available to the user and that other tools may be equally effective or more effective in terms of results, time, and cost.

When M&S is chosen as the methodology to be used, then further definition of M&S type is required. Options include (a) use of an existing (legacy) simulation as is, (b) modification of an existing (legacy) simulation to meet the user's requirements, or (c) development of a new simulation specifically focused on the user's requirements and objectives. Based on this decision, the model or simulation is implemented and applied to the defined problem. Results are integrated, presented to the application sponsor, and archived for future reference. Although Figure 3-2 reflects a linear process, considerable iteration occurs to refine the process as it progresses through the life cycle.

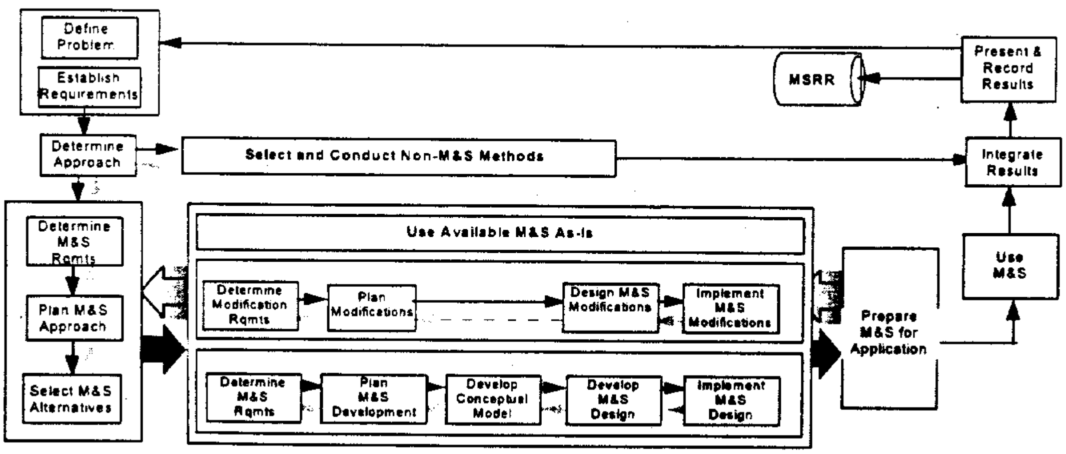

Figure 3-2. The M&S Life Cycle

The strategy selected will determine the detailed steps necessary to support implementation and application of the model or simulation. The steps associated with each of the three strategies are as follows.

3.3.1.1 Use Available Model or Simulation As Is

In this strategy the user elects to implement an existing (legacy) model or simulation without major modification. The decision to use a legacy model or simulation is generally based on either financial limitations or the user's level of comfort with the simulation, based on previous experience or lack of knowledge about alternative simulations. Since the user is ultimately responsible for the results produced by the selected model or simulation, user confidence is a prime motivator in model or simulation selection.

By accepting a legacy model or simulation, the user implicitly accepts its inherent underlying assumptions, limitations, and constraints. Unfortunately, because many legacy models and simulations have not undergone formal VV&A and have no documented conceptual model, the user may not have a clear understanding of the underlying assumptions, limitations, and constraints. Thus, it is most important for the analyst to map the results to the Measures of Effectiveness (MOEs) and Measures of Performance (MOPs) identified as part of the requirements definition stage.

3.3.1.2 Modify an Existing Model or Simulation

Although the use of a legacy simulation as is does occur, a far more common strategy is the modification of an existing (legacy) model or simulation to meet the user's requirements. This strategy essentially merges the legacy and new development concepts. The implementation steps associated with this strategy parallel those associated with new simulation development with one exception: the lack of formal conceptual model development. Since the foundation of the completed implementation rests on the existing code, an understanding of the original developer's intent or conceptual model is critical. The conceptual model definition includes its underlying assumptions, constraints, and limitations. Although the conceptual model is not formally identified in the modification process diagram (Figure 3-2), it is important that the individuals altering the simulation understand the original developer's intent as well as the current vision for merging the modified code with the existing code. The steps associated with this strategy are as follows (again note that iteration exists at all phases of development).

3.3.1.2.1 Determine Modification Requirements. The user-defined requirements are essential to the development and VV&A efforts. These requirements define the functionality (what the model does) and capability (how well the model does it) that the user requires of the model or simulation. These requirements serve as a framework against which the model or simulation is validated. A set of lower level software and system needs also are derived from the user's requirements. Associated with each requirement is a priority indicating its relative importance to the potential customer's needs. This ranking is a useful decision tool if time or cost constrain the extent of V&V that can be performed. When the model or simulation is to be modified, the higher level requirements focus on the customer's needs, but the lower level requirements address only those parts of the system or software to be changed.

The priorities associated with the user's requirements flow down to the software and system requirements and to the software and system design and implementation. Traceability of requirements through all stages of development helps ensure that the user's needs are being met in the implementation.

Once the developer's vision is established, the low-level requirements of the system and software are defined. Referred to as the Software Requirements Specification, these requirements define the hardware, software, and personnel needed to execute the model or simulation. The specification includes hardware and software for networks and protocols in distributed M&S. Commencement of final model coding before completing the M&S specification is not good practice and can lead to wasteful expenditure of resources and inappropriate code. Preliminary code prepared as part of the rapid prototyping software development approach and selected high-risk code developed in parallel with the specification to ensure feasibility for that element of code are not prohibited.

3.3.1.2.2 Plan Modifications. The planning phase of the process defines the roadmap for the development effort. Functions that support planning include the following.

- Definition of MOEs and MOPs
- Definition of scenarios
- Identification of resources and resource availability
- Definition of schedule

- Preliminary development of supporting plans such as federation testing, VV&A, configuration management, and quality assurance. In this instance, plans specify the modifications that are to be made and the approach that will be taken to make them.

3.3.1.2.3 Design Modifications. The outcome of the design phase is the developer's blueprint for the model or simulation. The design process has two primary components: the architectural system design, which addresses the hardware and software architecture, data structures, and interfaces, and the detailed software design, which addresses key elements of the software such as critical algorithms and data issues. Design features emphasize functionality, information flow, ordering of processes, and data accessibility. Any software elements defined in the M&S design are developed in accordance with contemporary standard software development procedures such as the ANSI/IEEE series or DoD standards. During the M&S design phase, the development plan will be updated to reflect more accurately management issues (tasks, schedule, and resources) to be addressed and analysis actions (scope, limitations, constraints, methodology, sources of data, testing, and acceptability criteria) to be taken. In this instance, the design will focus on the required modifications. Documentation that supports the original M&S design is extremely helpful to any modification effort. If the documentation does not exist, parts of it that are relevant to the specific application may need to be redeveloped to support the modification.

3.3.1.2.4 Implement Modifications. M&S implementation is the combination of computer code, processes, equipment, networks, operators, and personnel that compose the model or simulation. By maintaining connections among the requirements, the design, and the implementation, it is possible to identify the elements of the design or implementation that are affected by a given requirement. As requirements shift, these mappings help simplify the modification process.

3.3.1.2.5 Prepare for Application. The model or simulation is applied to a specific problem using resources developed during the design, construction, and test phase to satisfy objectives established during the planning and requirements phase. This phase does not begin until V&V has been completed.

3.3.1.2.6 Use Model or Simulation and Integrate Results. Once the model or simulation has been accredited, it is implemented. Output data are collected and results are analyzed, after action reviews are conducted and the accreditation report is prepared.

3.3.1.2.7 Present and Record Results. Results are forwarded to the decision maker according to established reporting requirements.

3.3.1.3 Develop a New Model or Simulation

In this strategy the user elects to build the model or simulation from scratch and defines specific requirements to which the model or simulation will be built. This approach allows the most effective integration of VV&A into the development process, as VV&A can be incorporated in the earliest stages and tightly coupled with each succeeding phase of development. The steps associated with this strategy mirror those associated with the modification of an existing simulation (see Section 3.3.1.2), with the addition of the definition of a formal conceptual model.

The conceptual model serves as a bridge between the defined requirements and the M&S design, providing the developer's interpretation of the requirements to which the model or simulation will be built. The conceptual model is a statement of assumptions, algorithms, and architecture that relates the elements of the model to one another (and to other models or simulations in federated simulation environments) for the model's or simulation's intended applications. The conceptual model also addresses the availability of appropriate, certified input data for the new model or simulation. The approach to developing the conceptual model should be iterative, allowing communication between the developer and the intended user. Failure to develop an adequate conceptual model before final design and implementation has been a major cause of past M&S inadequacies.

3.3.2 M&S Development Paradigms

The M&S life cycle defined in Section 3.3.1 is generic in nature and can be implemented in many different ways, including the waterfall, spiral, iterative, evolutionary, fountain, rapid application development, and model-test-model methods. Availability of resources, especially time, must be considered when selecting a development methodology or paradigm. When the time schedule is tight or compressed, the best method is the one that is familiar and simple to use. Newer, unfamiliar methods can be selected when learning time will not have a significant schedule impact.

Some of these approaches are discussed in the following paragraphs.

3.3.2.1 Waterfall Development Cycle

The Waterfall Development Cycle (Figure 3-3) is the more traditional development process for M&S. It is a structured, step-by-step functional development process that closes out each phase before starting the next. This structured process also facilitates

In-Process Reviews (IPR), Preliminary Design Reviews (PDR), and Critical Design Reviews (CDR) at the end of each step in the development. This structured review correlates the intent of the developers and the desire of the user. Before the next step proceeds, differences are resolved and approval by the cognizant authority obtained.

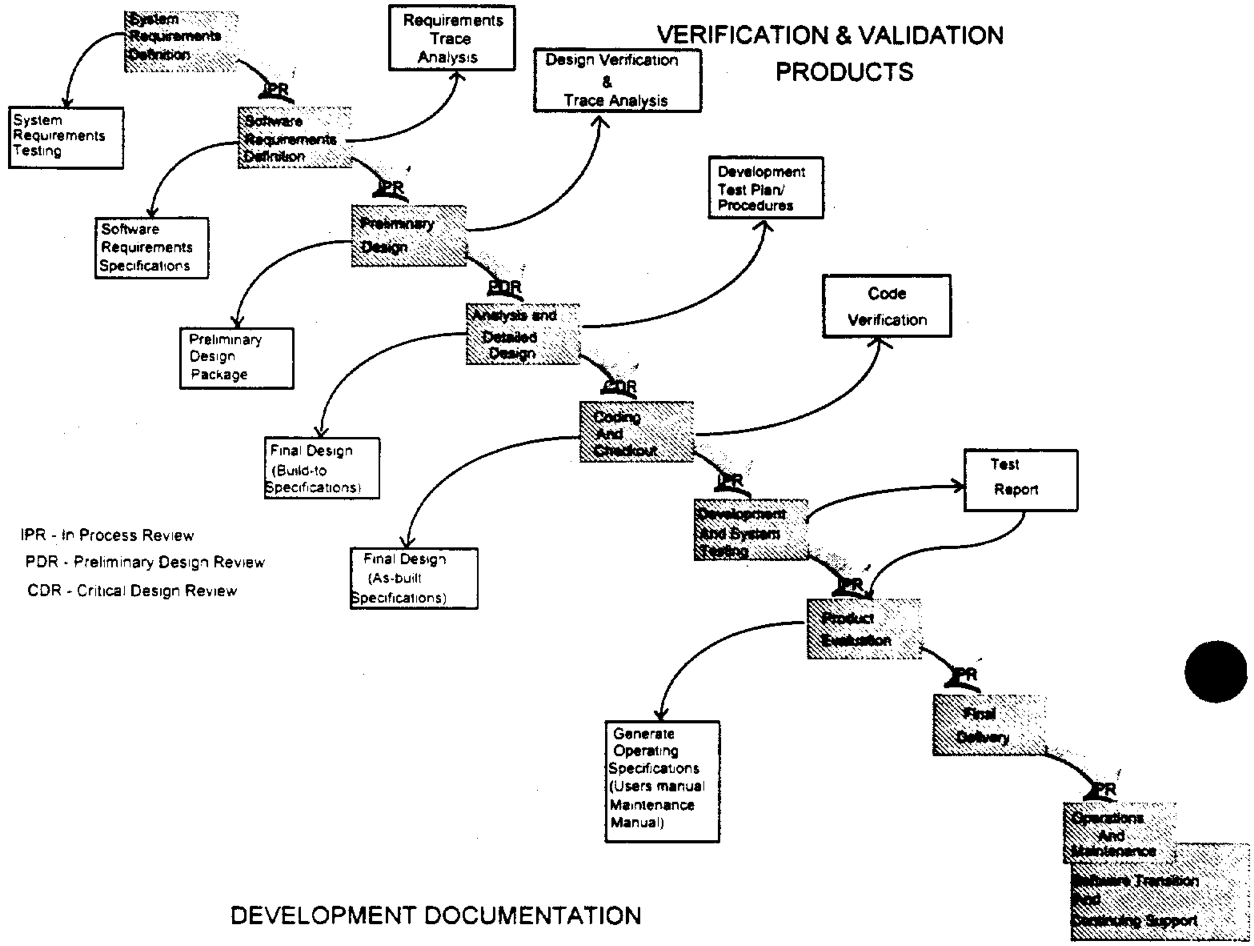

Figure 3-3. The Waterfall Development Cycle

Other characteristics of this process include the following:

- It encourages specification before building the system: requirements are defined before designing.
- It assesses the interaction of components before they are built: design before implementation.

- It enables the tracking of progress more accurately to uncover possible slippages early.
- It facilitates the generation of a series of documents that can be utilized later to test and maintain the system.

3.3.2.2 Spiral Development Cycle

The spiral software development cycle, shown in Figure 3-4, is an evolutionary prototyping methodology that is extremely useful when requirements are not well-defined.

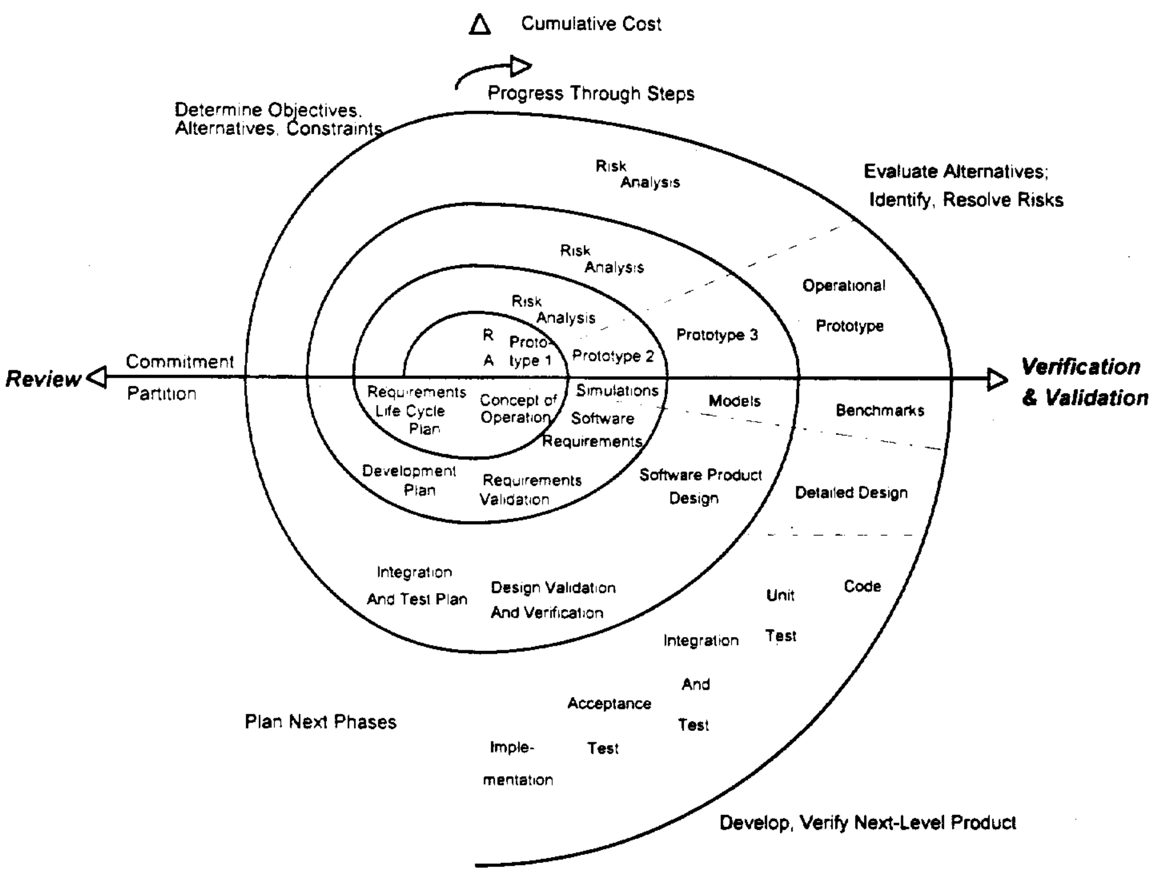

Figure 3-4. The Spiral Development Cycle

The spiral methodology employs an iterative process, with the first iteration beginning at the center of the spiral and working outward. A partial implementation of the system that meets the known or perceived requirements is constructed. The prototype is then employed and evaluated at the same time by its intended user in order to understand the full requirements better. The spiral model has four major activities:

- Planning—determining objectives, alternatives, and constraints of the development effort
- Risk analysis—analysis of the alternative approaches that could be employed and identification of risk
- Engineering—design and implementation of the model or simulation
- Customer evaluation—assessment of the resulting product

As defined in the spiral development process, *evolutionary* prototyping implies that requirements are not all known at the beginning and experiments with the operational system are needed to create a more useful product. *Incremental* development implies that most of the requirements are understood initially and are implemented in subsets of increasing capability. With this method, the developer is more apt to start implementation with those aspects of the system that are best understood and thus build on strengths.

3.3.2.3 High-Level Architecture Federation Development Process

As has been previously noted, the development of DoD's Common Technical Framework significantly affects the way in which M&S is used in DoD. The HLA is the central focus of the Common Technical Framework and offers a unique solution to building models, simulations, and federations by promoting interoperability and reuse. The emphasis is on providing those elements of federations that are common to all uses so they need not be rebuilt each time. These features include a run time infrastructure, rules, interface specifications, and object model templates. Technical documents are available that explain the details of these features; however, the following description of the HLA, illustrated in Figure 3-5, is intended to be as easy to understand as the material will allow!

The HLA can be applied to all three functional areas and can use all three M&S classes illustrated in Figure 3-1. HLA applications use federations of models and simulations, known as federates, which have been grouped together to solve a specific problem.

As with any application of M&S, the first step is for the application sponsor to define the problem statement and objectives. This step corresponds to the "Define Problem" and "Establish Requirements" boxes in the upper left corner of the M&S life-cycle diagram (Figure 3-2). The approach for an HLA application presumes the use of M&S to solve the problem that has been identified by the sponsor. The problem definition is used to generate specific M&S requirements, the approach that will be taken, and the selection of the model suite that will be used.

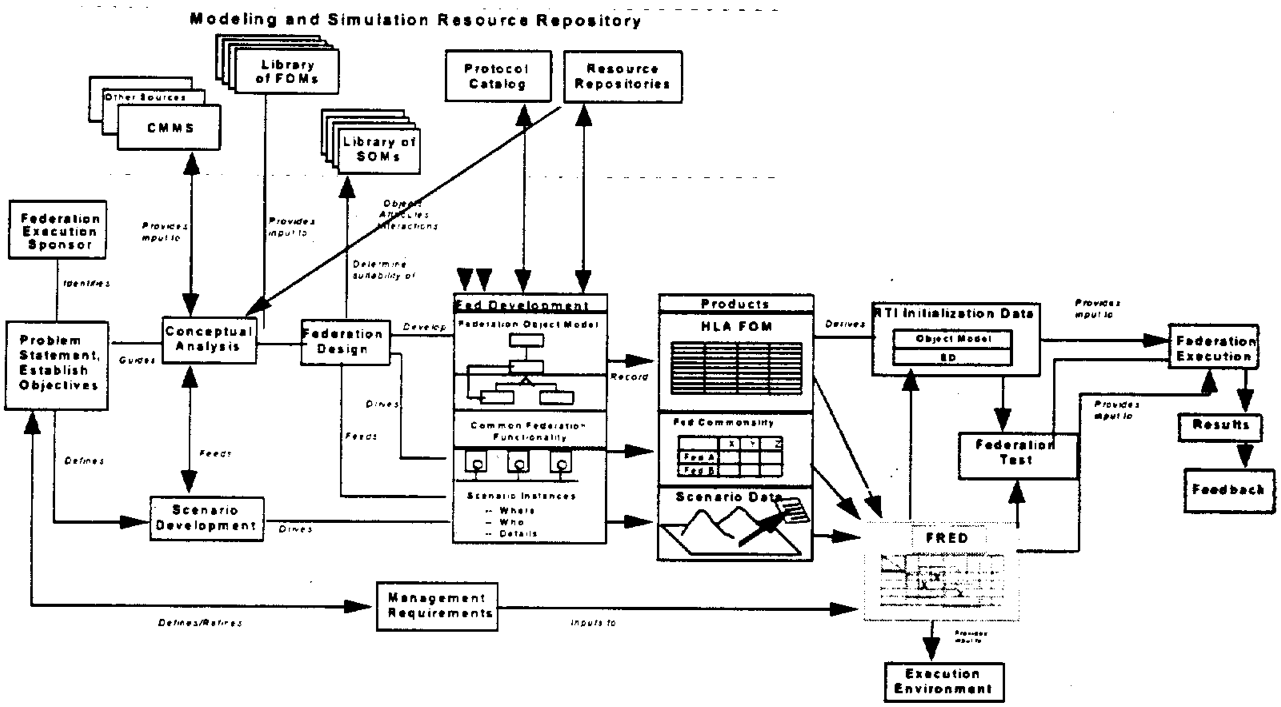

Figure 3-5. The HLA Federation Development Process

NOTE: Federation Object Models (FOMs), Simulation Object Models (SOMs), and Federation Required Execution Details (FRED) are discussed below.

The federation developers use high-level requirements to define a scenario in which the given problem is studied and solved. The scenario includes the major entities represented in the federation, a conceptual description of their capabilities, behavior, and interactions over time, and a specification of environmental factors and conditions. Scenario development is one of the key M&S requirements described in Figure 3-2.

The next step is a conceptual analysis that decomposes (breaks down) the scenario into conceptual-level components, which are usually expressed as objects and interactions. This step is part of the planning stage and precedes the development of a conceptual model. The result of this analysis is a conceptual model that provides a framework for the federation's design.

Conceptual analysis draws upon the Conceptual Models of the Mission Space (CMMS), which is the second of the three legs of the Common Technical Framework (the HLA and Data are the other two). CMMS are first abstractions of the real world; they capture basic information about entities, their actions, and interactions from a simulation-neutral viewpoint. CMMS content is validated by authoritative data sources from the warfighter community. The CMMS is based on the Uniform Joint Task List (UJTL).

The next step is the design of the federation itself. Although it would seem that this step would correspond directly to the "Develop M&S Design" step in the generic M&S development process, it also includes part of the conceptual modeling phase. The primary emphasis is the identification of the principal members of the federation and negotiation among these federates as to how the federation will be developed. Other tasks include defining the objects, attributes, and interactions that will be exchanged among federates; outlining specific responsibilities of each federate; and reviewing existing Federation Object Models (FOMs) and Simulation Object Models (SOMs) that may be re-used in the federation under development.

SOMs are descriptions of those key features, including objects, behaviors, and relationships, that an individual simulation brings to the federation negotiation table. The FOM is the superset of the SOMs that have been selected for use in a given federation. The FOM incorporates the definition of all the objects, interactions, state transitions, and communication flows that will occur within the federation. The FOM is the federation blueprint, an agreement between the federates concerning what will be built.

FOMs and SOMs are stored in the Modeling and Simulation Resource Repository (MSRR), which also includes data, metadata, models, simulations, and VV&A histories. In addition to FOMs and SOMs, the federation design also calls upon protocol catalogs that contain standard data definitions and formats. Protocol catalogs are, likewise, contained in the MSRR.

Simultaneous to federation design and part of the generic "Develop M&S Design" step is the development of FRED, the Federation Required Execution Details. In a nutshell, FRED is how the FOM works internally. It includes networking requirements, the physical connections that make the federation work, and the platforms and nodes of which the federation is composed.

The generic "Implement the M&S Design" step parallels the next step of federation development. In this step, the FOM, common simulation functionality, and data needed to support the federation scenario are developed collaboratively among the federates. Common simulation functionality comprises those tasks that all the federates need to do and can use the same thing to do it, such as a common clock, a common data base, or shared common algorithms that ensure a fair fight when the simulations run together.

Federation development also includes confirmation of each federate's responsibilities to each other. Relationships between objects are defined. Negotiations among the federates continue as to what attributes (planes fly) and level of functionality (how high) must be developed, incorporated, and maintained by the federates. The federates also must agree on object interaction protocols (how do tanks act around ground troops?) and common representations (which terrain data base will be used?).

The products from the federation development stage are the FOM, definition of common simulation functionality, and identification of scenario details.

Completing development is the Run Time Infrastructure, or RTI. The RTI is simply the physical implementation of the three big pieces of any HLA application: the rules, the interface specifications, and the object model template. These detailed documents were mentioned in the first paragraph of this section; they are available from the DMSO Web page where you probably found this guide!

The RTI needs data from the FOM and FRED to start up. Beginning as a clean slate, the RTI first takes "object model data" from the FOM. These are simply the tables of data that will be exchanged among the federates. The other data taken from FRED are the execution details of how the federation runs, how information is passed, and who gets what messages. RTI initialization is equivalent to the "Prepare M&S for Application" box in the generic process.

The federation is now ready to be tested. There are two kinds of tests, HLA compliance testing and federation functional testing. The first asks if information gets passed correctly within the federation when it is connected to the RTI. The second tests the logical interactions between the federates, checking that the information that is passed makes sense.

Finally, the federation is run and the results are analyzed to obtain a solution to the problem that was specified at the very beginning. This step is the point of the process, to answer the questions posed and provide the decision maker with a solution.

3.3.2.4 Distributed Interactive Simulation Exercise Development Process

DIS is a government and industry initiative to define an infrastructure for linking simulations of various types at multiple locations to create a realistic, complex, virtual environment for the simulation of interactive activities. (See Figure 3-11.) This infrastructure brings together platforms from the Military Services and systems built by various vendors using different technologies for different purposes and permits them to interoperate. DIS exercises support a mixture of virtual entities with computer-controlled behavior (computer-generated forces), virtual entities with live operators (human-in-the-loop simulators), live entities (operational platforms, test and evaluation systems), and constructive entities (automated simulations, wargaming). DIS draws heavily on experience derived from the Simulator Networking (SIMNET) program developed by the Defense Advance Research Projects Agency (DARPA), adopting many of SIMNET's basic concepts and heeding lessons learned from those experiences.

The DIS exercise development process illustrated in Figure 3-6 consists of the five major activities or phases summarized in the following paragraphs.

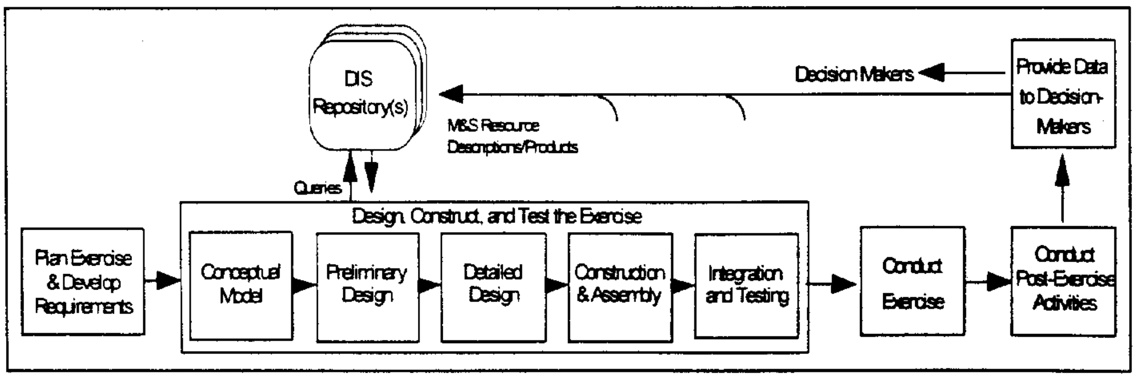

Figure 3-6. The DIS Exercise Management and Feedback Life Cycle

3.3.2.4.1 Plan Exercise and Develop Requirements. This phase includes a number of functions that support proper planning:

- Determining MOEs, MOPs, and exit criteria applicable to the exercise

- Developing support plans (e.g., VV&A plan, VV&C plan)
- Defining exercise environment (e.g., weather, climate, electromagnetic conditions, oceanographic features)
- Determining the mix of simulation forces among live, virtual, and constructive categories
- Determining simulation resources available
- Determining technical and exercise support personnel required
- Developing requirements and network interface specifications.

These same functions support the development of VV&A plans.

3.3.2.4.2 Design, Construct, and Test the Exercise. In this phase, the exercise is developed to meet the requirements specified during the planning phase. This phase consists of five steps:

- Conceptual model—The conceptual model represents the exercise architect's understanding of the exercise requirements and purpose. It serves as the foundation for the design and development of the exercise configuration.
- Preliminary design—The conceptual model is translated into a high-level design of the exercise. An architecture is created to show the participating components, their interfaces, behavior, and control structure.
- Detailed design—The design model and architecture generated in the previous step are elaborated to support the complete definition of all required functions, data flow, and behavior, including communication data-rate requirements and data-latency limitation requirements.
- Construction and assembly—The existing DIS components are assembled and new components are developed.
- Integration and testing—This step is usually performed as an incremental process, starting with a minimum number of components and connectivity and building until operational status is achieved. Testing occurs to determine if requirements and performance criteria are met.

Verification and validation activities are conducted during and following each step and results must be accepted by the exercise manager before proceeding. Section 3.4.4.3 provides additional information on the DIS VV&A process.

3.3.2.4.3 Conduct the Exercise. The exercise is conducted using resources developed during the design, construction, and test phase to satisfy objectives established during the planning phase. This phase does not begin until exercise verification and validation has been completed and exercise configuration has been accredited.

3.3.2.4.4 Conduct the Post-Exercise Activity. This activity includes the collection and processing of output data, analysis of results, after action review (AAR), and preparation of exercise documentation.

3.3.2.4.5 Provide Results to Decision Makers. Exercise results are reported to the decision makers according to the reporting requirements of the exercise.

3.3.2.5 Aggregate-Level Simulation Protocol Exercise Development Process

The Joint Training Confederation (JTC) is an integrated network of distributed interoperable simulations used by the Commanders in Chief (CINCs) and subordinate commands in joint training exercises to identify wartime capability and readiness issues. The ALSP Program supports the JTC by providing the simulation architecture, protocols, and software that integrate the individual Service campaign-level simulations into a single environment. The JTC is revised annually to reflect key aspects of air, land, and maritime warfare operations and training requirements identified by the CINCs.

The JTC development cycle begins with the existing JTC capabilities, simulations, and test tools. Feedback from the CINCs and Services identifies deficiencies and recommends functional improvements to the participating simulations or changes in the ALSP architecture to increase training realism or to improve efficiency.

3.4 The VV&A Process in the M&S Life Cycle

The VV&A process is an integral part of the M&S life cycle (Figure 3-7). The primary purpose of the VV&A effort is to establish the credibility of the model or simulation. Much like building a body of evidence in a court case, the VV&A

agent derives and accumulates data that will support a judgment or accreditation decision regarding the acceptability of the model or simulation for a given application.

A secondary function of VV&A is to support risk mitigation. By identifying potential errors and problems early in the development process, verification and validation efforts aid in the development of an accurate and cost-effective model or simulation.

3.4.1 Process Description

The following paragraphs describe the seven steps of the VV&A process, which are grouped in the box entitled "Conduct Verification, Validation, and Accreditation" in the lower right corner of Figure 3-7.

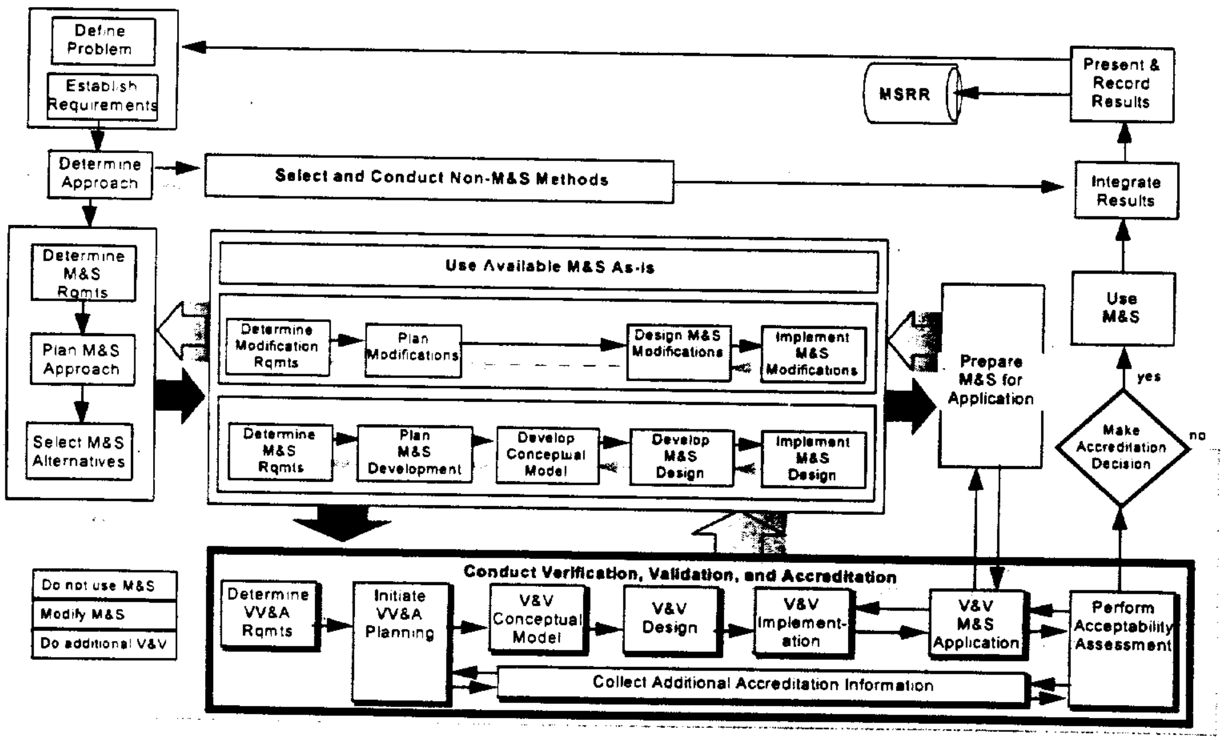

Figure 3-7 The Generic VV&A Process

3.4.1.1 Determine VV&A Requirements

Once the method for implementing a model or simulation has been chosen (legacy, modify, or build new), requirements must be defined by which the success of the VV&A effort will be judged. VV&A requirements include determining the level of effort for the VV&A process and techniques that will be used, as well as logistic factors such as the identification of the V&V agent, number of workhours required, hardware and software needs, and an estimate of overall VV&A costs.

3.4.1.2 Initiate VV&A Planning

The focus of each plan (see Chapter 6 for more information on plans) is to identify the tasks required in a manner that matches and complements the M&S plan, requirements, resources, and timelines. Each plan is adapted to address the requirements and constraints of the M&S application and covers critical issues, while allowing flexibility for adjustment and refinement.

Formal guidance and requirements are collected and reviewed to determine the constraints under which the model V&V; Verification, Validation, and Certification (VV&C); VV&C, and accreditation efforts will operate and appropriate evaluation techniques and measures are identified. Necessary tools and resources are further identified and specific activities scheduled. Initially, the plans are developed as drafts or working documents that evolve as the application takes shape. When new information is available or changes occur, the plans are reviewed and updated as appropriate.

3.4.1.3 V&V the Conceptual Model

In Chapter 1 "conceptual model verification" was loosely defined as "Did I build the thing right?" and "conceptual model validation" as "Did I build the right thing?" Verification satisfies the functional requirements, validation the fidelity requirements. Both the conceptual model and its V&V must be documented. The documentation explains why (or why not) the assumptions, algorithms, modeling concepts, anticipated data availability, and architecture of the conceptual model are expected to provide an acceptable representation of the subject modeled for intended application of the model or simulation. Any interactions expected with other models or simulations (as in a federation) must be taken into account. Conceptual model verification and validation should occur before further M&S development to avoid the potential pitfall of inaccurately representing the system and not meeting the proposed requirements. Errors caught at this early stage of development are easier and less expensive to fix.

3.4.1.4 V&V the Design

As it is constructed, the M&S design is verified against the conceptual model to ensure that it accurately reflects the validated concept and associated requirements. The M&S design has an associated V&V plan, which addresses management (tasks, schedule, and resources) and analysis (scope, limitations, constraints, methodology, sources of data, testing, and acceptability criteria) actions for V&V during M&S development. In some cases, an Independent Verification and Validation (IV&V) plan may be appropriate. (See Chapter 1 for a discussion of the relevance of IV&V.)

3.4.1.5 V&V the Implementation

Once the implementation of the design is completed in code, the results of the model or simulation are formally (i.e., documented) reviewed. Responses of the model or simulation are compared against known or expected behavior from the subject it represents to ascertain that the M&S responses are sufficiently accurate for the intended use. The developer of a model with stochastic processes is expected to provide guidance regarding the number of iterations required for statistically significant results.

3.4.1.6 V&V the Application

Once the model or simulation is ready to be run, the application context needs to be verified and validated. This includes such housekeeping tasks as ensuring that the appropriate platforms are being used and that operators and humans-in-the-loop are properly trained.

3.4.1.7 Perform Acceptability Assessment

This step reviews the information collected during the V&V assessment of the model or simulation for use in the intended application. This is the final step before deciding to accredit and use the model or simulation for the given purpose. Documentation that supports the acceptability assessment includes a comparison of the application M&S requirements to the simulation's capabilities and limitations; model or simulation development and use history; model or simulation operating requirements and cost;

implications of the model's or simulation's limitations and constraints for use in this application; and recommendations for changes to allow the model or simulation to be used for the application or to reduce application risk. (Chapter 6 contains additional guidance for preparing the *Acceptability Assessment Report*.)

3.4.2 A Note on Tailoring

A VV&A effort must be cost-effective, responsive, and sufficient to succeed. To maintain a balance between application requirements and real-world constraints, the VV&A process should be tailored to fit the purpose of the application and the type(s) of simulation(s) involved. Tailoring, the selection of verification and validation techniques (see Chapter 4) based on requirements and resource availability, is done as part of the VV&A planning process to determine the most appropriate and cost-effective ways to address the application requirements and acceptability criteria.

3.4.3 VV&A As Applied to High-Level Architecture

The HLA federation development life cycle shown in Figure 3-8 has been modified to reflect the interaction with VV&A. The HLA Baseline Definition document includes a section that discusses many of the VV&A aspects discussed in the following paragraphs.

As discussed earlier, the initial tasks of stating the problem and establishing requirements are combined in the HLA process diagram (Figure 3-5). Determining VV&A requirements naturally are included in this process.

VV&A planning is initiated in the Conceptual Analysis stage of HLA federation development. It uses the products of Scenario Development to determine the degree of V&V that is required to ensure the accurate representation of major entities and their interactions. Environmental conditions also must be verified and validated to ensure consistency with conceptual intent and real-world accuracy at a level that is appropriate to the intended use of the model.

V&V of the conceptual model includes three major portions of the federation development process (speckled overlay). Conceptual Analysis, Federation Design, and portions of Federation Development all involve Conceptual Model V&V. The definition of objects and interactions which results from the Conceptual Analysis stage requires V&V to ensure that these objects and interactions are accurately represented.

Identification of the federates and their individual responsibilities is one focus of Federation Design. Here, V&V plays a major role in checking the V&V history of the federates and determining the additional V&V that is required to make those simulations credible for the purposes of the federation. Emphasis is placed on the realistic portrayal of federate capabilities in carrying out proposed responsibilities within the federation.

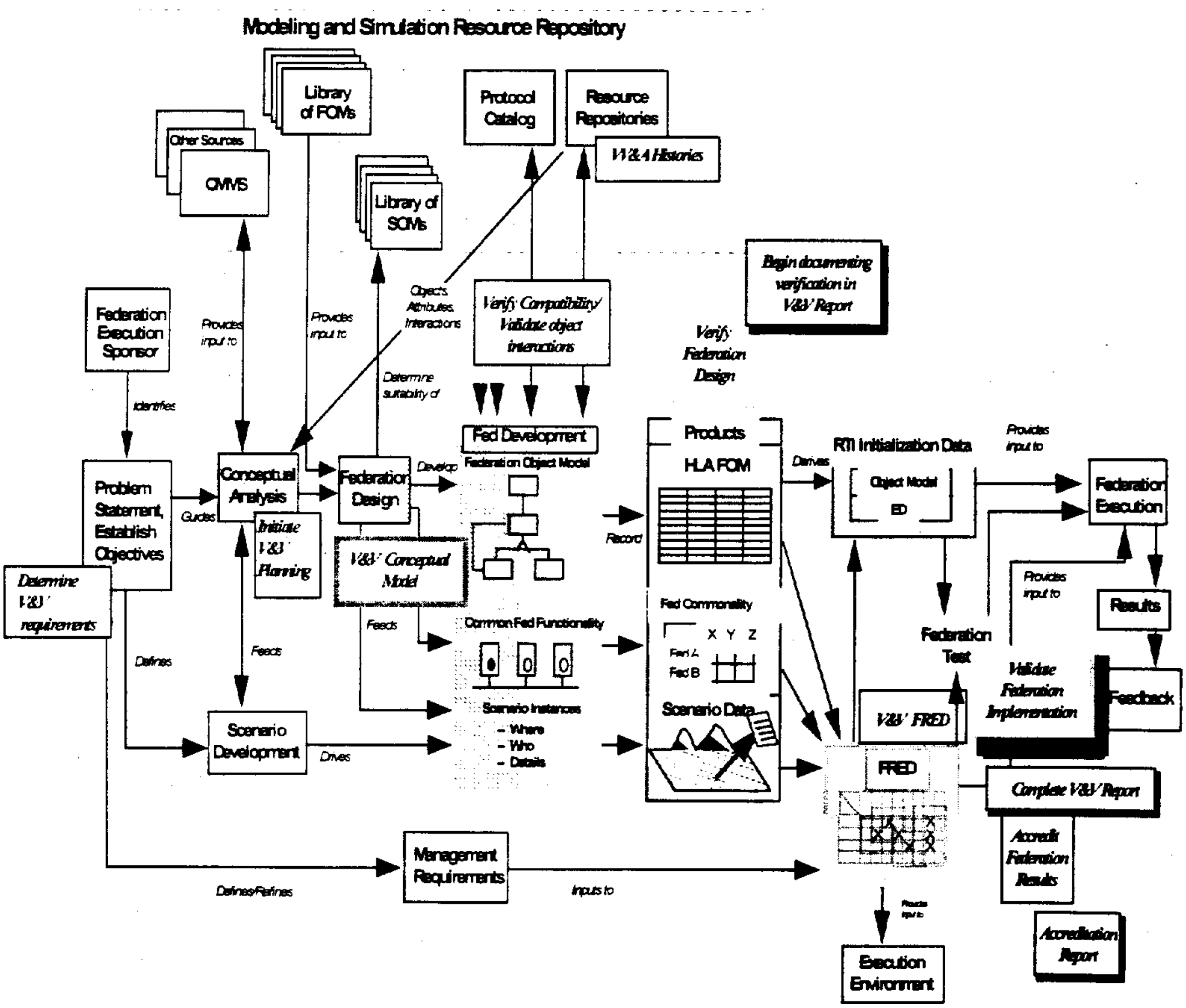

Figure 3-8. The VV&A Process in the HLA Federation Development Life Cycle

Another objective of Federation Design is to identify potential opportunities for reuse of existing FOMs and SOMs. As discussed in Section 3.3.2.3, FOMs and SOMs describe the capabilities of federations and federates to assist other users in determining their suitability for new applications. Both *FOMs and SOMs need to be validated against the federations and simulations they represent* to ensure consistency in the descriptions provided with the actual federation or federate.

The Federation Development stage is the final area where V&V of the Conceptual Model occurs. Federation Development bridges the V&V function across the Conceptual Model to V&V of the Federation Design (striped overlay). Conceptual and design activities include FOM development, as well as identification of common functionalities, data requirements, object relationships, common syntax, and semantics. As design features become more detailed, V&V is performed to ensure that they accurately reflect the intent of the conceptual design. MSRR resources also are retrieved during Federation Development. These resources include histories of previous VV&A efforts on federates and federations that are similar in application or that may be considered for application or modification in the current federation. Information from the MSRR is verified to ensure compatibility and to validate object interactions across federates.

Design V&V extends from the Federation Development stage to include part of the FRED. The FRED describes the way the FOM works internally to the federation. Network requirements, physical connections, and delineation of platforms and nodes must all be verified against the developer's specifications. HLA compliance testing meets much of this V&V requirement.

V&V of the implementation of the federation involves the products of the federation development process, portions of FRED, the RTI initialization data, and the federation test (orange/shaded overlay). Federation documents generated during development offer excellent traceability for V&V activities. RTI initialization data show the physical implementation of the rules, interface specifications, and object model. These data, as well as those obtained from FRED, serve as valuable conduits through which V&V is performed to ensure that the implementation of the federation accurately reflects the intended design.

Federation Testing includes both HLA compliance testing and federation functional testing. The former ensures that, when the federation is connected to the RTI, the interface specifications are handled properly and information is passed correctly. This correlates directly to verification, which checks the implementation against the developer's conceptual description and specifications. A similar parallel can be drawn between functional testing, which looks for logical interactions and ensures that the information that is passed makes sense, and validation, which tests the credibility of the implementation against the real world.

Figure 3-8 also indicates the points in the process at which reports and documentation of the VV&A effort should occur. These documents are an integral part of the overall application of M&S.

3.4.4 Migration of the Generic VV&A Process to Different Types of Applications

3.4.4.1 Legacy M&S

Figure 3-9 illustrates the generic VV&A process modified to include the two options of using an existing model as is or modifying it to meet new user requirements.

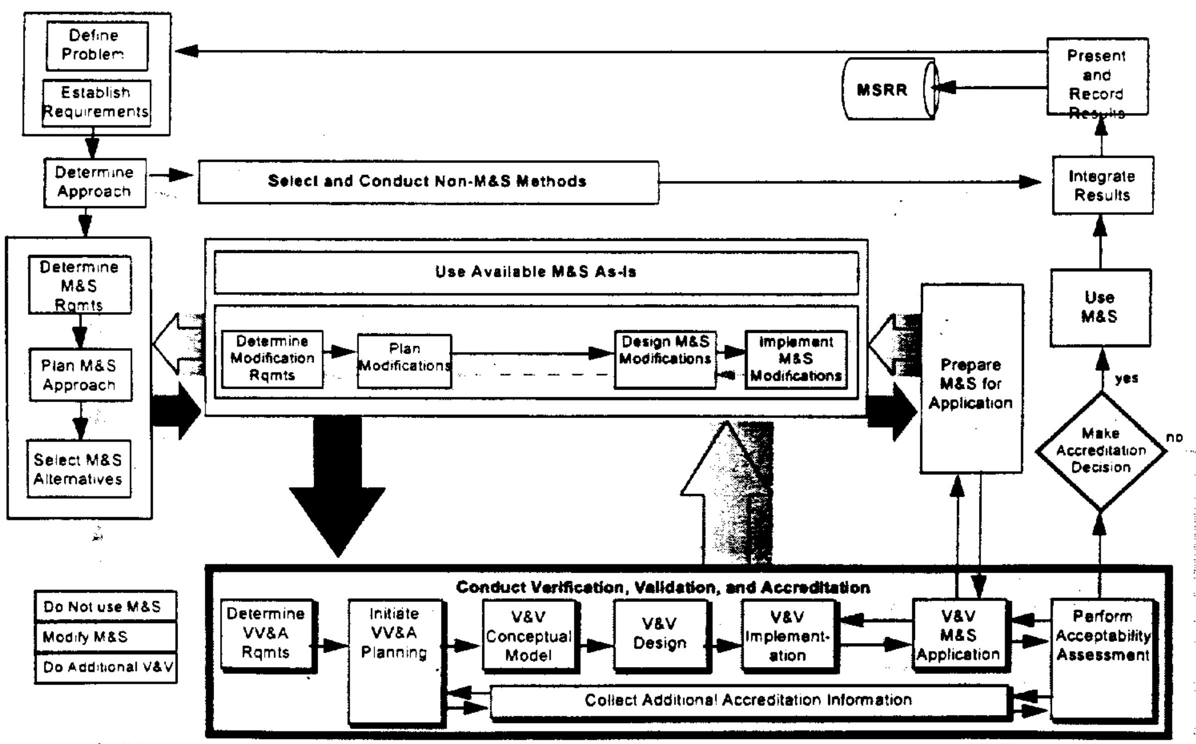

Figure 3-9. The VV&A Process in the Legacy M&S Life Cycle

3.4.4.2 New M&S

Figure 3-10 again alters the generic VV&A process to include only those sections pertinent to new M&S development.

3.4.4.3 Distributed Interactive Simulation

The DIS nine-step VV&A process (see Figure 3-11) was accepted by a consensus agreement of the DIS VV&A Subgroup of the DIS Workshop, which represents the training functional area community for distributed simulation. It is discussed in detail in the DIS Recommended Practices Documents being developed for DIS VV&A and DIS Exercise Control. The VV&A process parallels the DIS exercise development process. A major assumption of the DIS process is that each individual component has undergone some level of VV&A (e.g., according to a given Service's policy) independent of a DIS exercise configuration. Each of the nine steps is defined in the following paragraphs.

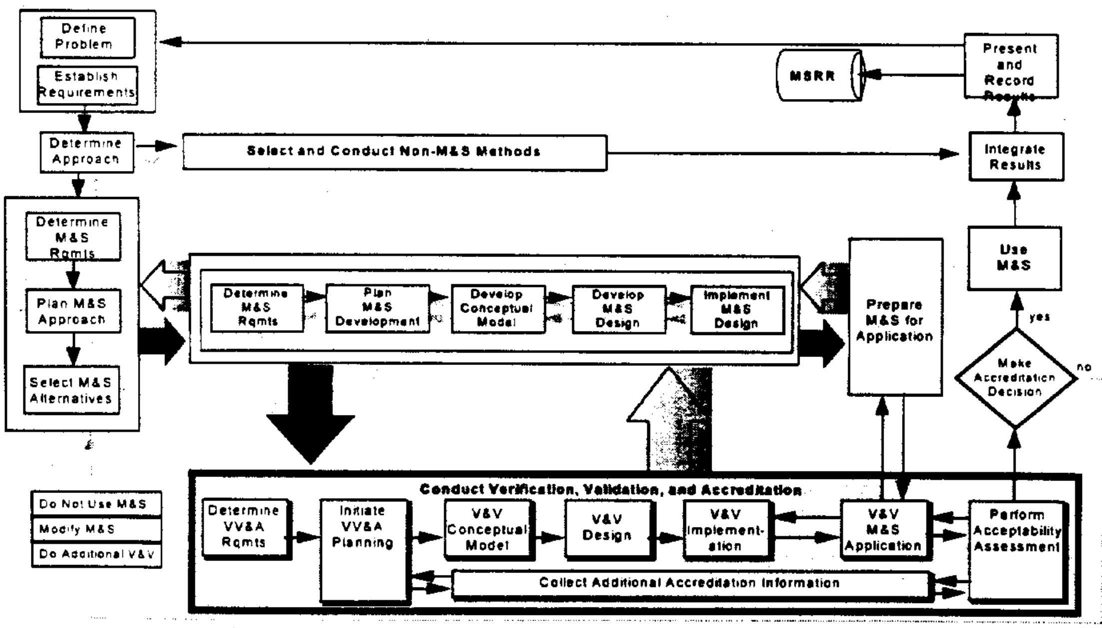

Figure 3-10. The VV&A Process in the New M&S Life Cycle

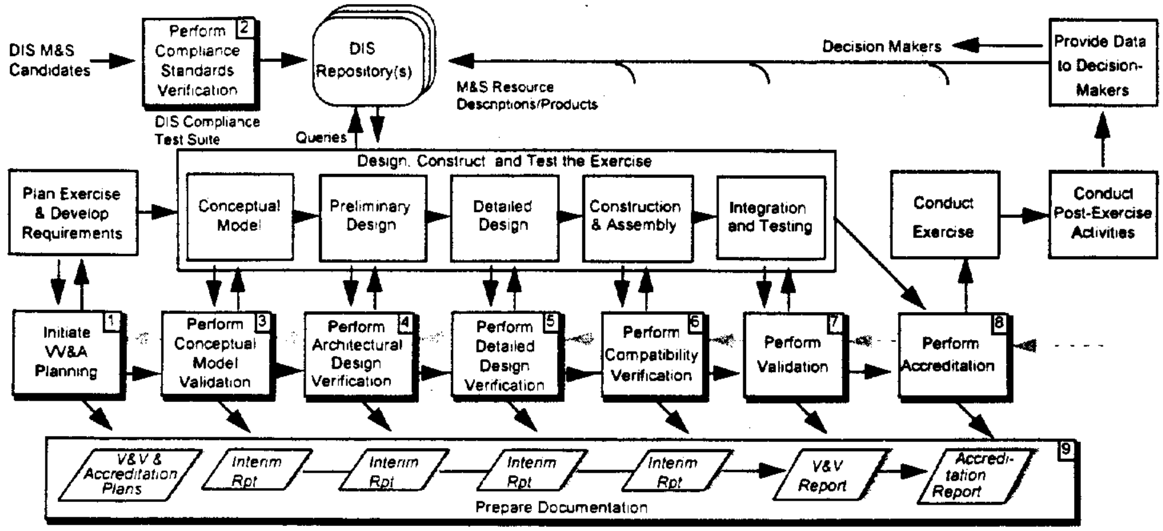

Figure 3-11. The VV&A Process in the DIS Exercise Management and Feedback Life Cycle

3.4.4.3.1 Develop VV&A Plans. VV&A planning begins in the earliest stages of DIS exercise development when exercise plans are being produced and the associated exercise requirements, e.g., the type of systems that need to be represented, the level of fidelity that is required, are being defined. At this point, the VV&A and testing plans are conceptualized and drafted, and the exercise requirements are validated.

3.4.4.3.2 Verify Standards. At this stage, proposed DIS components (i.e., model, simulation, or simulator; live, virtual, or constructive) are tested to verify that they can communicate adequately using the DIS Protocol Data Units (PDU). This step can occur before or during DIS exercise development. The Institute for Simulation in Training (IST) in association with the Simulation, Training, and Instrumentation Command (STRICOM) have developed a compliance test suite to assist in testing for protocol compliance.

3.4.4.3.3 Perform Conceptual Model Validation. During this phase, the conceptual model is validated against the exercise requirements. The conceptual model offers an initial configuration of DIS compatible components that satisfies the exercise requirements. Traceability of requirements to the conceptual model and preliminary design is stressed. This step is iterated until a conceptual model that satisfactorily meets the required objectives is defined.

3.4.4.3.4 Perform Architectural Design Verification. This phase of VV&A is tied to the development of the preliminary design or conceptual model for the exercise. Information contained in a DIS repository about candidate DIS components, their associated component level VV&A history, and fidelity characteristics can assist in making design decisions. The conceptual model or preliminary design is verified for correctness and completeness.

3.4.4.3.5 Perform Detailed Design Validation. In the detailed design phase, the preliminary design or conceptual model discussed in Steps 3 and 4 is expanded to a detailed level. Validation at this stage ensures that detailed design is correct and complete and maintains traceability to the requirements.

3.4.4.3.6 Perform Compatibility Verification. At this point, the compatibility of the components within the DIS exercise configuration is verified.

3.4.4.3.7 Perform Exercise Validation. This phase of the V&V process examines how well the DIS exercise configuration represents the behavior, appearance, performance, fidelity constraints, and interoperability levels of the intended application.

3.4.4.3.8 Perform Accreditation. The V&V conducted for the exercise is reviewed by the accrediting authority (i.e., exercise user or sponsor) and an accreditation decision for formal acceptance is made.

3.4.4.3.9 Prepare VV&A Reports. Descriptions and results of the VV&A effort are documented and funneled to the DIS Repository as evidence of VV&A activity and for potential use in future DIS exercises.

As with the HLA development and VV&A processes, the DIS exercise development is directly mapped to the nine-step VV&A process and those processes defined in the generic life cycle and VV&A descriptions of Sections 3.2 and 3.4.

3.4.4.4 Aggregate-Level Simulation Protocol

VV&A activities are integrated into the development cycle for each year's confederation and apply only to the ALSP protocols and software. (See Figure 3-12.) The activities focus on ensuring interoperability of component simulations within the confederation framework and on run time performance. Each simulation in the JTC has been approved by a participating Service or Agency and is considered accredited for use in the JTC. Improvements to the participating simulations, however, are coordinated with the Services and the ALSP office to ensure continued compatibility for future JTCs.

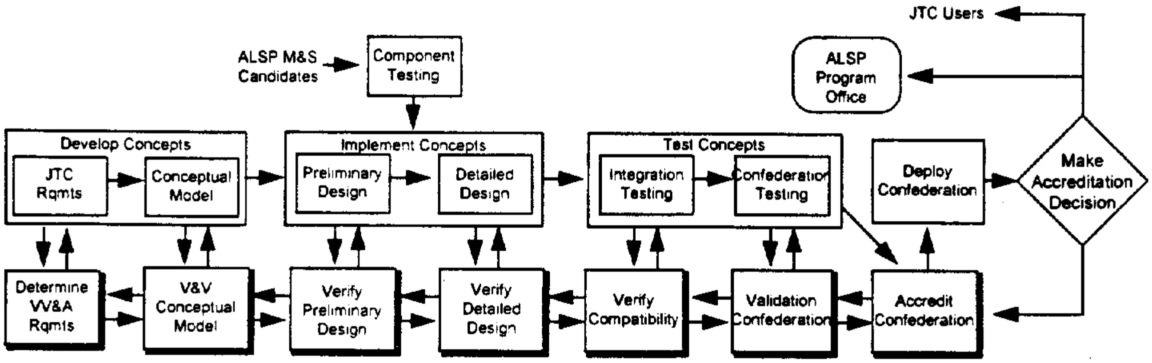

Figure 3-12. The VV&A Process in the ALSP Life Cycle

The V&V activities include reviews of each design and document by the ALSP Review Panel. Methods range from formal structured walkthroughs to informal briefings with the level of formality commensurate with the priority or novelty of the concept and the estimated risk associated with its integration.

Chapter 4—Techniques

This chapter presents Verification and Validation (V&V) techniques and provides guidelines for their use. Seventy-six V&V techniques and eighteen statistical techniques that can be used for model validation are described. Most of these techniques are derived from software engineering; the remaining are specific to the modeling and simulation field. The selected software V&V techniques applicable to Modeling and Simulation (M&S) V&V are presented in terms understandable by an M&S technical person. Some software V&V techniques are modified for use in M&S V&V. The term *testing* is used frequently in this chapter in referring to the implementation of these techniques. V&V requires the testing of the model or simulation to assess its credibility. Finally, where possible, supporting texts are referenced so that more detailed descriptions of the techniques may be obtained by the interested reader.

4.1 Verification and Validation Techniques

Figure 4-1 shows a taxonomy that lists V&V techniques in four categories: informal, static, dynamic, and formal. The use of mathematical and logical formalism in each category increases from informal to formal, from left to right. The complexity also increases as the category becomes more formal.

It should be noted that some of the categories presented in Figure 4-1 possess similar characteristics and, in fact, include techniques that overlap from one category to another. A distinct difference between each classification exists, however, as will be evident in the discussion.

4.1.1 Informal V&V Techniques

Informal techniques are among the most commonly used. They are called informal because their tools and approaches rely heavily on human reasoning and subjectivity without stringent mathematical formalism. The *informal* label does not imply, however, a lack of structure or formal guidelines in their use. In fact, these techniques are applied using well-structured approaches under formal guidelines, and they can be very effective if employed properly.

Verification and Validation

Informal
- Audit
- Desk Checking
- Face Validation
- Inspections
- Reviews
- Turing Test
- Walkthroughs

Static
- Cause-Effect Graphing
- Control Analysis
  - Calling Structure
  - Concurrent Process
  - Control Flow
  - State Transition
- Data Analysis
  - Data Dependency
  - Data Flow
- Fault/Failure Analysis
- Interface Analysis
  - Model Interface
  - User Interface
- Semantic Analysis
- Structural Analysis
- Symbolic Evaluation
- Syntax Analysis
- Traceability Assessment

Dynamic
- Acceptance Testing
- Alpha Testing
- Assertion Checking
- Beta Testing
- Bottom-Up Testing
- Comparison Testing
- Compliance Testing
  - Authorization
  - Performance
  - Security
  - Standards
- Debugging
- Execution Testing
  - Monitoring
  - Profiling
  - Tracing
- Fault/Failure Insertion Testing
- Field Testing
- Functional (Black-Box) Testing
- Graphical Comparisons
- Interface Testing
  - Data
  - Model
  - User
- Object-Flow Testing
- Partition Testing
- Predictive Validation
- Product Testing
- Regression Testing
- Sensitivity Analysis
- Special Input Testing
  - Boundary Value
  - Equivalence Partitioning
  - Extreme Input
  - Invalid Input
  - Real-Time Input
  - Self-Driven Input
  - Stress
  - Trace-Driven Input
- Statistical Techniques
- Structural (White-Box)
  - Branch
  - Condition
  - Data Flow
  - Loop
  - Path
  - Statement
- Submodel/Module Testing
- Symbolic Debugging
- Top-Down Testing
- Visualization/Animation

Formal
- Induction
- Inference
- Logical Deduction
- Inductive Assertions
- Lambda Calculus
- Predicate Calculus
- Predicate Transformation
- Proof of Correctness

Figure 4-1. A Taxonomy of Verification and Validation Techniques

4.1.1.1 Audit

An audit is undertaken to assess how adequately the application of M&S is conducted with respect to established plans, policies, procedures, standards, and guidelines. The audit also seeks to establish traceability within the simulation. When an error is identified, it should be traceable to its source via its audit trail. The process of documenting and retaining sufficient evidence about the substantiation of accuracy is called an *audit trail* (Perry, 1995).

Auditing is carried out periodically through a mixture of meetings, observations, and examinations (Hollocker, 1987). Audit is a staff function and serves as the "eyes and ears of management" (Perry, 1995, p. 26). In Verification, Validation, and Accreditation (VV&A), auditing is performed by the VV&A agent throughout the development life cycle for a new model or simulation or during modifications made to legacy models and simulations.

4.1.1.2 Desk Checking

Desk checking (also called *self-inspection*) is the process of intensely examining work to ensure its correctness, completeness, consistency, and clarity. It is considered to be the very first step in V&V and is particularly useful for the early stages of development. To be effective, desk checking should be conducted carefully and thoroughly, preferably by another person, because it is usually difficult to see one's own errors (Adrion *et al.*, 1982). Syntax review, cross-reference examination, convention violation assessment, detailed comparison to specifications, code reading, control flowgraph analysis, and path sensitizing are all be conducted as part of desk checking (Beizer, 1990).

4.1.1.3 Face Validation

The project team members, potential users of the model, and people knowledgeable about the system of interest use their estimates and intuition to compare model and system behaviors subjectively under identical input conditions and judge whether the model and its results are reasonable (Hermann, 1967).

This technique is regularly cited in V&V efforts within DoD. It is one of the terms and techniques most commonly misused. Face validation is useful mostly as a preliminary approach to validation in the early stages of development. Except for a model that is mature and has an extensive, well-documented VV&A history, viable V&V efforts generally must use additional techniques.

4.1.1.4 Inspections

A team with four to six members inspects any M&S development phase such as M&S requirements definition, conceptual model design, or M&S detailed design. To inspect M&S design, for example, the team might consist of a moderator who manages the inspection team and provides leadership; a reader who narrates the M&S design and leads the team through the inspection process; a recorder who produces a written report of detected faults; a designer who represents the design developer; an implementer who translates the M&S design into an executable form; and a VV&A agent.

An inspection goes through five distinct phases: overview, preparation, inspection, rework, and follow-up (Schach, 1996). In Phase I, the designer summarizes the M&S design to be inspected. Characteristics such as problem definition, application requirements, and the specifics of software design are introduced and related documentation is distributed to all participants to study. In Phase II, the team members prepare individually for the inspection by examining the documents in detail. The success of the inspection rests heavily on the conscientiousness of the team members in their preparation. The moderator arranges the inspection meeting with an established agenda and chairs it in Phase III. The reader narrates the M&S design documentation and leads the team through the inspection process. The inspection team is aided during the fault-finding process by a checklist of queries. The objective is to find and document the faults, not to correct them. The recorder prepares a report of detected faults immediately after the meeting. In Phase IV, the designer resolves all faults and problems identified in the report. In the final phase, the moderator ensures that all faults and problems have been resolved satisfactorily. All changes must be examined carefully to ensure that no new errors have been introduced as a result of a fix.

Inspections have major differences from walkthroughs, described in Section 4.1.1.7. Briefly, a walkthrough is less formal, has fewer steps, and does not use a checklist to guide or a written report to document the team's work. By comparison, an inspection is a five-step, formalized process. The inspection team uses the checklist approach for uncovering errors. The inspection process takes much longer than a walkthrough; however, the extra time is justified because an inspection is a powerful and cost-effective way of detecting faults early in the M&S development life cycle (Ackerman *et* *al.*, 1983; Beizer, 1990; Dobbins, 1987; Knight and Myers, 1993; Perry, 1995; Schach, 1996).

4.1.1.5 Reviews

The review is conducted similar to the inspection and walkthrough, except that the review team also involves managers. The review is intended to give management, such as the M&S proponent or the M&S application sponsor, evidence that the M&S development process is being carried out according to stated application objectives and to evaluate the model or simulation in light of development standards, guidelines, and specifications. As such, the review is a higher level technique than the inspection or walkthrough.

Each review team member examines the M&S documentation before the review. (Given the management positions of the team members, documentation needs to be less technical and more oversight-oriented than in an inspection. There also must be less material to examine if the V&V agent expects the team to prepare satisfactorily for the review.) The team then meets to evaluate the model or simulation relative to specifications and standards, recording defects and deficiencies. The review team may be given a set of indicators to measure such as (a) appropriateness of the problem definition and M&S requirements, (b) adequacy of all underlying assumptions, (c) adherence to standards, (d) modeling methodology used, (e) model representation quality, (f) model structure, (g) model consistency, (h) model completeness, and (i) documentation. (A checklist prepared by the V&V agent [not the developer!] is particularly useful in focusing management on the key points and in guiding the review.) The result of the review is a document portraying the events of the meeting, deficiencies identified, and review team recommendations. Appropriate action then may be taken to correct any deficiencies.

As opposed to inspections and walkthroughs, which concentrate on assessing correctness, reviews seek to ascertain that tolerable levels of quality are being attained. The review team is more concerned with model or simulation design deficiencies and deviations from stated model or simulation development policy than it is with the intricate line-by-line details of the implementation. This does not imply that the review team is not concerned with discovering technical flaws in the model or simulation, only that the review process is oriented toward the early stages of the M&S development life cycle (Hollocker, 1987; Perry, 1995; Sommerville, 1996; Whitner and Balci, 1989).

4.1.1.6 Turing Test

Turing test is based upon the expert knowledge of people about the system of interest. The experts are presented with two sets of output data, one obtained from the model and one from the system, under the same input conditions. Without identifying the data set, the experts are asked to differentiate between the two. If they succeed, they are asked to describe the differences. Their response provides valuable feedback for correcting model representation. If they cannot differentiate between the two, confidence in the model's validity is increased (Schruben, 1980; Turing, 1963; Van Horn, 1971).

4.1.1.7 Walkthroughs

A typical structured walkthrough team consists of a coordinator, often the V&V agent, who organizes, moderates, and follows up the walkthrough activities; a presenter, who is usually the model or simulation developer; a scribe who documents the events of the walkthrough meetings; a maintenance oracle who focuses on long-term implications; a standards bearer who assesses adherence to standards; the accreditation agent who reflects the needs and concerns of the accrediting authority; and other reviewers such as the model or simulation project manager and auditors. Except for the model or simulation developer, none of the team members should be involved directly in the development effort.

The main thrust of the walkthrough is to detect and document faults; it is *not* performance appraisal of the development team. This point must be made to everyone involved so that full cooperation is achieved in discovering errors.

The coordinator schedules the walkthrough meeting, distributes the walkthrough material to all participants well in advance to allow for careful preparation (again, critical to the success of the effort!), and chairs the meeting. During the meeting, the presenter narrates the walkthrough documents. (The V&V agent may wish to ascertain the level of preparation of the team members at the beginning of the meeting to ensure that materials have been read beforehand and that team members are not relying on the presenter's walkthrough of the material to obtain the information and insight needed for a meaningful discussion.) The coordinator encourages questions and discussion to uncover any faults (Adrion *et al.*, 1982; Deutsch, 1982; Myers, 1978, 1979; Yourdon, 1985).

The reader is encouraged to re-read Sections 4.1.1.4 and 4.1.1.5 on inspections and reviews to ensure a full understanding of the differences among these three techniques.

4.1.2 Static V&V Techniques

Static V&V techniques assess the accuracy of the static model design and source code. Static techniques do not require machine execution of the model, but mental execution can be used. The techniques are very popular and widely used, and many automated tools are available to assist in the V&V process. The simulation language compiler is itself a static V&V tool.

Static V&V techniques can reveal a variety of information about the structure of the model, the modeling techniques used, data and control flow within the model, and syntactical accuracy (Whitner and Balci, 1989).

4.1.2.1 Cause-Effect Graphing

Cause-effect graphing addresses the question of what causes what in the model representation. It first identifies causes and effects in the system being modeled and then examines their representation in the model specification. For example, in the simulation of a traffic intersection, the following causes and effects may be identified: (a) the change of a light to red immediately causes the vehicles in the traffic lane to stop, (b) an increase in the duration of a green light causes a decrease in the average waiting time of vehicles in the traffic lane, and (c) an increase in the arrival rate of vehicles causes an increase in the average number of vehicles at the intersection.

As many causes and effects as possible are listed, and the semantics are expressed in a cause-effect graph. The graph is annotated to describe special conditions or impossible situations. Once the cause-effect graph has been constructed, a decision table is created by tracing back through the graph to determine combinations of causes that result in each effect. The decision table then is converted into test cases with which the model is tested (Myers, 1979; Pressman, 1996; Whitner and Balci, 1989).

4.1.2.2 Control Analysis

Control analysis techniques consist of calling structure analysis, concurrent process analysis, control flow analysis, and state transition analysis.

Calling structure analysis is used to assess model accuracy by identifying who *calls* whom and who is *called* by whom. The *who* could be a procedure, subroutine, function, method, or a submodel within a model. For example, inaccuracies caused by message passing (e.g., sending a message to a nonexistent object) in an object-oriented model can be revealed by analyzing the specific messages that invoke an action and the actions that messages invoke (Miller *et al.*, 1995).

Concurrent process analysis is especially useful for parallel (Fujimoto, 1990, 1993; Page and Nance, 1994) and distributed simulations. If a simulation executes on a single computer with a single processor (CPU), it is referred to as a *serial (sequential) simulation*. If a single computer with multiple processors is used to execute the simulation model, then the simulation is said to be a *parallel simulation*. If multiple single-processor computers are used to execute the simulation model, then the simulation is said to be a *distributed simulation*.

Model accuracy is assessed by analyzing the overlap or simultaneous execution of actions executed in parallel or across distributed simulations. Such analysis can reveal synchronization and time management problems (Rattray, 1990).

Control flow analysis requires the graphing of the model, in which conditional branches and model junctions are represented by nodes and the model segments between such nodes are represented by links (Beizer, 1990). A node of the model graph usually represents a logical junction where the flow of control changes, whereas an edge represents the junction that assumes control. This technique examines sequences of control transfers and is useful for identifying incorrect or inefficient constructs within model representation.

State transition analysis identifies the finite number of states through which the model execution passes. A state transition diagram, which shows how the model transitions from one state to another, is created. Model accuracy is assessed by analyzing the conditions under which a state change occurs. This technique is especially effective for those M&S applications created under the activity scanning, three-phase, and process interaction conceptual frameworks (Balci, 1988).

4.1.2.3 Data Analysis

The data analysis category of V&V techniques consists of Data Dependency Analysis and Data Flow Analysis. These techniques are used to ensure that (a) proper operations are applied to data objects (e.g., data structures, event lists, linked lists), (b) the data used by the model are properly defined, and (c) the defined data are properly used (Perry, 1995).

Data dependency analysis determines which variables depend on other variables (Dunn, 1984). For parallel and distributed simulations, the data dependency knowledge is critical for assessing the accuracy of synchronization across multiple processors.

Data flow analysis assesses model accuracy with respect to the use of model variables. This assessment is classified according to the definition, referencing, and unreferencing of variables (Adrion *et al.*, 1982), i.e., when variable space is allocated, accessed, and deallocated. A data flowgraph is constructed to aid in the data flow analysis. The nodes of the graph represent statements and corresponding variables. The edges represent control flow.

Data flow analysis can be used to detect undefined or unreferenced variables (much as in static analysis) and, when aided by model instrumentation, can track minimum and maximum variable values, data dependencies, and data transformations during model execution. It is also useful in detecting inconsistencies in data structure declaration and improper linkages among submodels or federates (Allen and Cocke, 1976; Whitner and Balci, 1989).

4.1.2.4 Fault/Failure Analysis

Fault (incorrect model component) and failure (incorrect behavior of a model component) analysis uses model input-output transformation descriptions to identify how the model logically *might* fail. The model design specification is examined to determine if any failures logically could occur, in what context, and under what conditions. Such examinations often lead to identification of model defects (Miller *et al.*, 1995).

4.1.2.5 Interface Analysis

Interface analysis consists of model interface analysis and user interface analysis. These techniques are especially useful for verification and validation of interactive and distributed simulations.

Model interface analysis examines submodel-to-submodel interfaces within a model, or federate-to-federate interfaces within a federation, and determines if the interface structure and behavior are sufficiently accurate.

User interface analysis examines the user-model interface and determines if it is human engineered to prevent errors during the user's interactions with the model. It also assesses how accurately this interface is integrated into the overall model or simulation.

4.1.2.6 Semantic Analysis

Semantic analysis is conducted by the simulation programming language compiler and determines the modeler's intent as reflected by the code. The compiler describes the content of the source code so the modeler can verify that the original intent is reflected accurately.

The compiler generates a wealth of information to help the modeler determine if the true intent is translated accurately into the executable code: (a) *symbol tables,* which describe the elements or symbols that are manipulated in the model, function declarations, type and variable declarations, scoping relationships, interfaces, and dependencies; (b) *cross-reference tables,* which describe called versus calling routines (where each data element is declared, referenced, and altered), duplicate data declarations (how often and where occurring), and unreferenced source code; (c) *subroutine interface tables,* which describe the actual interfaces of the caller and the called; (d) *maps,* which relate the generated runtime code to the original source code; and (e) *pretty printers* or *source code formatters,* which reformat the source listing on the basis of its syntax and semantics, clean pagination, highlighting of data elements, and marking of nested control structures (Whitner and Balci, 1989).

4.1.2.7 Structural Analysis

Structural analysis examines the model structure and determines if it adheres to structure principles. It is conducted by constructing a control flowgraph of the model structure and examining the graph for anomalies, such as multiple entry and exit points, excessive levels of nesting within a structure, and questionable practices such as the use of unconditional branches (i.e., GOTOs).

Yucesan and Jacobson (1992, 1996) apply the theory of computational complexity and show that the problem of verifying structural properties of M&S applications is difficult to solve. They illustrate that modeling issues such as accessibility of states, ordering of events, ambiguity of model specifications, and execution stalling are problems for which general design techniques do not produce efficient solutions.

4.1.2.8 Symbolic Evaluation

Symbolic evaluation assesses model accuracy by exercising the model using symbolic values rather than actual data values for input. It is performed by feeding symbolic inputs into the submodel or federate and producing expressions for the output that are derived from the transformation of the symbolic data along model execution paths. Consider, for example, the following function:

```
function jobArrivalTime(arrivalRate,currentClock,randomNumber)
      lag = -10
      Y = lag * currentClock
      Z = 3 * Y
      if Z < 0 then
            arrivalTime = currentClock - log(randomNumber) /
                  arrivalRate
      else
            arrivalTime = Z - log(randomNumber) / arrivalRate
      end if
      return arrivalTime
end jobArrivalTime
```

In symbolic execution, lag is substituted in Y resulting in Y = `(-10*currentClock)`. Substituting again, Z is found to be equal to `(-30*currentClock)`. Since `currentClock` is always zero or positive, an error is detected in that Z will never be greater than zero, and the "if-then-else" statement is unnecessary.

When unresolved conditional branches are encountered, a path is chosen to traverse. Once a path is selected, execution continues down the new path. At some point, the execution evaluation will return to the branch point and the previously unselected branch will be traversed. All paths eventually are taken.

The result of the execution can be represented graphically as a symbolic execution tree (Adrion *et al.*, 1982; King, 1976). The branches of the tree correspond to the paths of the model. Each node of the tree represents a decision point in the model and is labeled with the symbolic values of data at that juncture. The leaves of the tree are complete paths through the model and depict the symbolic output produced.

Symbolic evaluation assists in showing path correctness for all computations regardless of test data and is also a great source of documentation, but it has the following disadvantages: (a) the execution tree can explode in size and become too complex as the model grows; (b) loops cause difficulties although inductive reasoning and constraint analysis may help; (c) loops make thorough execution impossible because all paths must be traversed; and (d) complex data structures may have to be excluded because of difficulties in symbolically representing particular data elements within the structure (Dillon, 1990; King, 1976; Ramamoorthy *et al.*, 1976).

4.1.2.9 Syntax Analysis

Syntax analysis is carried on by the simulation programming language compiler to ensure that the mechanics of the language are applied correctly (Beizer, 1990).

4.1.2.10 Traceability Assessment

Traceability assessment is used to match, one to one, the elements of one form of the model to another. For example, the elements of the system as described in the requirements specification are matched one to one to the elements of the model or simulation design specification. Unmatched elements *may* reveal either unfulfilled requirements or unintended design functions (Miller *et al.*, 1995).

4.1.3 Dynamic V&V Techniques

Dynamic V&V techniques require model execution; they evaluate the model based on its execution behavior. Most dynamic V&V techniques require *model instrumentation,* the insertion of additional code (probes or stubs) into the executable model to collect information about model behavior during execution. Probe locations are determined manually or automatically based on static analysis of the model's structure. Automated instrumentation is accomplished by a preprocessor that analyzes the model's static structure (usually via graph-based analysis) and inserts probes at appropriate places.

Dynamic V&V techniques usually are applied in three steps. In Step 1, the executable model is instrumented. In Step 2, the instrumented model is executed; in Step 3, the model output is analyzed and dynamic model behavior is evaluated.

For example, consider the worldwide air traffic control and satellite communication object-oriented visual M&S application created by using the Visual Simulation Environment (Balci *et al.*, 1995) in Figure 4-2. The model can be instrumented in Step 1 to record the following information every time an aircraft enters into the coverage area of a satellite: (a) aircraft tail number; (b) time; (c) aircraft's longitude, latitude, and altitude; and (d) satellite's position and identification number. In Step 2, the model is executed and the information collected is written to an output file. In Step 3, the output file is examined to reveal discrepancies and inaccuracies in model representation.

4.1.3.1 Acceptance Testing

Acceptance testing is conducted by either the M&S application sponsor and the sponsor's VV&A agents or the developer's quality control group in the presence of the sponsor's representatives. The model is operationally tested with the actual hardware and data to determine whether all requirements specified in the legal contract are satisfied (Perry, 1995; Schach, 1996).

4.1.3.2 Alpha Testing

Alpha testing is the operational testing of the initial version of the complete model by the developer at an in-house site uninvolved with the model development (Beizer, 1990).

4.1.3.3 Assertion Checking

An assertion is a statement that should hold true as the simulation executes. Assertion checking is a verification technique that checks what is happening against what the modeler assumes is happening to guard against potential errors. The assertions are placed in various parts of the model to monitor execution. They can be inserted to hold true globally, for the whole model; regionally, for some submodels; locally, within a submodel; or at entry and exit of a submodel.

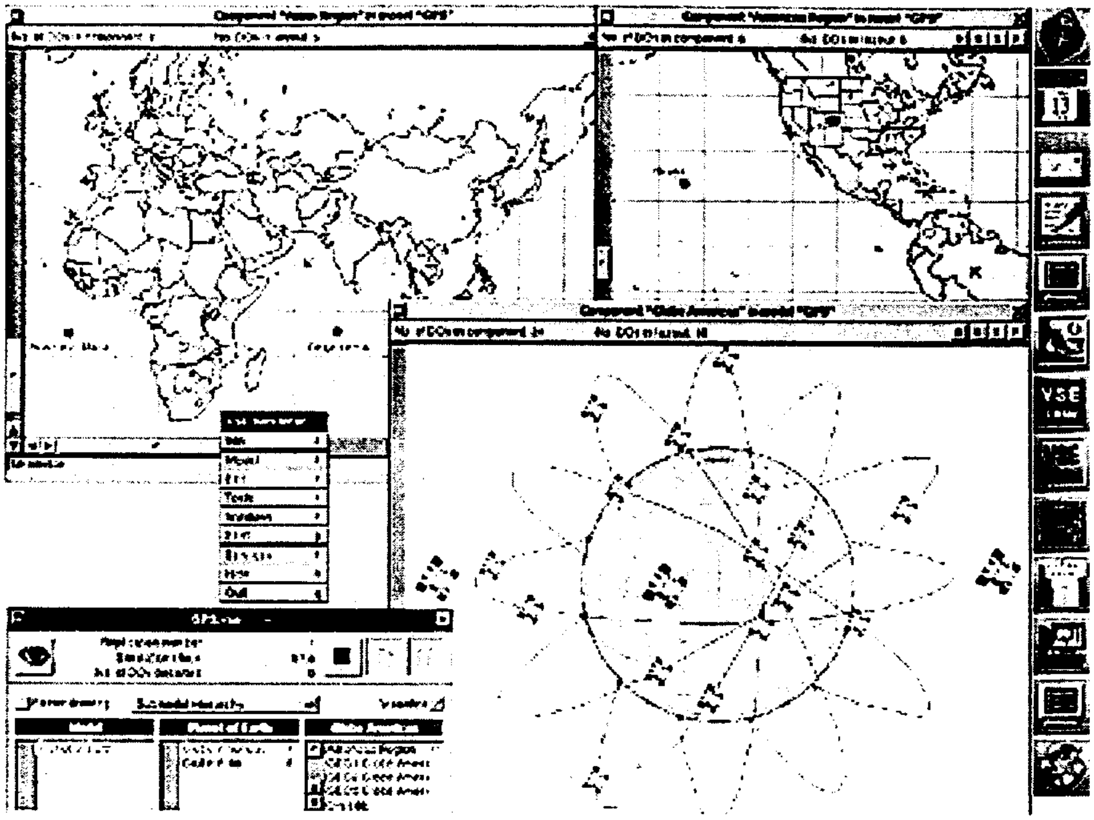

Figure 4-2. Visual Simulation of Global Air Traffic Control and Satellite Communication. (Reprinted from Balci *et al.*, 1995.)

Consider, for example, the following pseudo-code (Whitner and Balci, 1989):

```
Base := Hours * PayRate;
Gross := Base * (1 + BonusRate);
```

In just these two simple statements, several assumptions are being made. It is assumed that `Hours`, `PayRate`, `Base`, `BonusRate`, and `Gross` are all non-negative. The following asserted code can be used to prevent execution errors caused by incorrect values entered by the user:

```
Assert Local (Hours > 0 and PayRate > 0 and BonusRate > 0);
Base := Hours * PayRate;
Gross := Base * (1 + BonusRate).
```

Assertion checking also prevents structural model inaccuracies. For example, the model in Figure 4-2 can contain assertions such as (a) a satellite communicates with the correct ground station, (b) an aircraft's tail number matches its type, and (c) an aircraft's flight path is consistent with the official airline guide.

Clearly, assertion checking serves two important needs: (a) it verifies that the model is functioning within its acceptable domain, and (b) the assertion statement documents the intentions of the modeler. Assertion checking, however, degrades model performance, forcing the modeler to choose between execution efficiency and accuracy. If the execution performance is critical, the assertions should be turned off but kept permanently in code to provide both documentation and means for maintenance testing (Adrion *et al.*, 1982).

4.1.3.4 Beta Testing

Beta testing refers to the developer's operational testing of the first-release version of the complete model at a beta user site under realistic field conditions (Miller *et al.*, 1995).

4.1.3.5 Bottom-Up Testing

Bottom-up testing is used with bottom-up model development. In bottom-up development, model construction starts with the simulation's routines at the base level, i.e., the ones that cannot be decomposed further, and culminates with the submodels at the highest level. As each routine is completed, it is tested thoroughly. When routines with the same parent, or submodel, have been developed and tested, the routines are integrated and their integration is tested. This process is repeated until all submodels and the model as a whole have been integrated and tested. The integration of completed submodels need not wait for all submodels at the same level to be completed. Submodel integration and testing can be, and often is, performed incrementally (Sommerville, 1996).

Some of the advantages of bottom-up testing are (a) it encourages extensive testing at the routine and submodel levels; (b) because most well-structured models consist of a hierarchy of submodels, much may be gained by bottom-up testing; (c) the smaller the submodels and the more cohesion within the model, the easier and more complete its testing will be; and (d) it is particularly attractive for testing distributed models and simulations.

Major disadvantages of bottom-up testing include (a) individual submodel testing requires drivers, more commonly called test harnesses, which simulate the calling of the submodel and passing test data necessary to execute the submodel; (b) developing harnesses for every submodel can be quite complex and difficult; (c) the harnesses may themselves contain errors; and (d) bottom-up testing faces the same cost and complexity problems as does top-down testing (see Section 4.1.3.26).

4.1.3.6 Comparison Testing

Comparison testing (also known as back-to-back testing) may be used when more than one version of a model or simulation representing the same system is available for testing (Pressman, 1996; Sommerville, 1996). For example, different simulations may have been developed by the different Services to simulate the same military combat aircraft. (The development of the High-Level Architecture (HLA), however, is intended to reduce greatly such redundant model development in favor of fewer simulations at less cost to DoD.) All simulations built to represent exactly the same system are run with the same input data and the model outputs are compared. Differences in the outputs reveal problems with model accuracy. The major disadvantage to this technique is the lack of information that generally exists about the validity of the other models. If two models both were written with a specific, unnoticed error in the code, the results might agree but would still be invalid.

4.1.3.7 Compliance Testing

Compliance testing compares the simulation to required security and performance standards. These techniques are particularly useful for testing federations of distributed and interactive models and simulations. Compliance testing methods for HLA compliance have been developed and are available from DMSO.

Authorization testing tests how accurately different levels of security access authorization are implemented in the simulation and how properly they comply with established rules and regulations. The test can be conducted by attempting to execute a classified model within a federation or by using classified input data to run a simulation without proper authorization (Perry, 1995).

Performance testing simply tests whether all performance characteristics are measured and evaluated with sufficient accuracy and if all established performance requirements are satisfied (Perry, 1995).

Security testing tests whether all security procedures are implemented correctly and properly. For example, penetrating the simulation while it is running and breaking into classified components such as secure databases can be attempted. Security testing evaluates the adequacy of protective procedures and countermeasures (Perry, 1995).

Standards testing substantiates that the M&S application is developed with respect to the required standards, procedures, and guidelines.

4.1.3.8 Debugging

Debugging is an iterative process that uncovers errors or misconceptions that cause the model's failure and defines and carries out the model changes that correct the errors. This iterative process consists of four steps. In Step 1, the model is tested, revealing the existence of errors (bugs). Given the errors detected, the cause of each error is determined in Step 2. In Step 3, the model changes necessary to correct the detected errors are identified and are carried out in Step 4. Step 1 is re-executed immediately after Step 4 to ensure successful modification, because a change correcting an error may create another one. This iterative process continues until no errors are identified in Step 1 after sufficient testing (Dunn, 1987).

4.1.3.9 Execution Testing

Execution testing consists of monitoring, profiling, and tracing techniques. These techniques collect and analyze execution behavior data to reveal model representation errors.

Execution monitoring reveals errors by examining low-level information about activities and events that take place during model execution. It requires the instrumentation of a model or simulation to gather data to provide activity- or event-oriented information about the model's dynamic behavior. For example, the model in Figure 4-2 can be instrumented to monitor the arrivals and departures of aircraft within a particular city, and the results can be compared with the official airline guide to judge model validity. The model also can be instrumented to provide other low-level information such as the number of late arrivals, the average passenger waiting time at the airport, and the average flight time between two locations.

Execution profiling reveals errors by examining high-level information (profiles) about activities and events that take place during model execution. It requires the instrumentation of an executable model to gather data to present profiles about the model's dynamic behavior. For example, the model in Figure 4-2 can be instrumented to produce histograms of aircraft departure times, arrival times, and passenger check-out times at an airport.

Execution tracing reveals errors by reviewing the line-by-line execution of a simulation. It requires the instrumentation of an executable model to trace the model's line-by-line dynamic behavior. The model in Figure 4-2 can be instrumented to record all aircraft arrival times at a particular airport. Then, the trace data can be compared with the official airline guide to assess model validity.

The major disadvantage of the tracing technique is that execution of the instrumented model may produce a large volume of trace data too complex to analyze. To overcome this problem, the trace data can be stored in a data base and the modeler can analyze it using a query language (Fairley, 1975, 1976).

4.1.3.10 Fault/Failure Insertion Testing

This technique inserts a fault (incorrect model component) or a failure (incorrect behavior of a model component) into the model and observes whether the model produces the invalid behavior as expected. Unexplained behavior may reveal errors in model representation.

4.1.3.11 Field Testing

Field testing places the model in an operational situation and collects as much information as possible for model validation. It is especially useful for validating models of military combat systems. Field testing conducted as part of the test and evaluation process is particularly important within DoD system acquisition. It is a major element of VV&A conducted during the development of new weapons systems and platforms. Although it is usually difficult, expensive, and sometimes impossible to devise meaningful field tests for complex systems, their use wherever possible helps both the project team and decision makers develop confidence in the model (Shannon, 1975; Van Horn, 1971). The greatest disadvantage of field testing is the lack of adequate test resources to produce statistically significant results. Often, simulation runs augment live test data in the development and decision processes.

4.1.3.12 Functional Testing

Functional testing (also called *black-box testing*) assesses the accuracy of model input-output transformation. It is applied by feeding inputs (test data) to the model and evaluating the accuracy of the corresponding outputs.

It is virtually impossible to test all input-output transformation paths for a reasonably large and complex simulation, because the number of those paths could be in the millions. Therefore, the objective of functional testing is to increase confidence in model input-output transformation accuracy as much as possible rather than to claim absolute correctness.

The generation of test data is a crucially important but very difficult task. The law of large numbers does not apply here. Successfully testing the model under 1,000 input values (test data) does not imply high confidence in model input-output transformation accuracy just because of the large number. Instead, the number 1,000 should be compared with the number of allowable input values to determine the percentage of the model input domain that is covered in testing. The more the model input domain is covered in testing, the more confidence is gained in the accuracy of the model input-output transformation (Howden, 1980; Myers, 1979).

4.1.3.13 Graphical Comparison

Graphical comparison is a subjective, inelegant, and heuristic, yet quite practical approach, especially useful as a preliminary step to model V&V. The graphs of values of model variables over time are compared with the graphs of values of system variables to investigate characteristics such as similarities in periodicities, skewness, number, and location of inflection points; logarithmic rise and linearity; phase shift; trend lines; and exponential growth constants (Cohen and Cyert, 1961; Forrester, 1961; Miller, 1975; Wright, 1972).

4.1.3.14 Interface Testing

Interface testing (also known as *integration testing*) tests the data, model, and user interfaces. Interface testing is more rigorous than the interface analysis discussed in Section 4.1.2.5.

Data interface testing assesses the accuracy of data entered into the model or derived from the model during execution. All data interfaces are examined to substantiate that all aspects of data input and output are correct. This form of testing is particularly important for those simulations in which the inputs are read from a data base or the results are stored in a data base for later analysis. The model's interface to the data base is examined to ensure correct importing and exporting of data (Miller *et al.*, 1995). Data interface testing is key to the relationship between the VV&A effort and the corresponding Verification, Validation, and Certification (VV&C) of data effort.

Model interface testing detects model representation errors created as a result of submodel-to-submodel or federate-to-federate interface errors or invalid assumptions about the interfaces. It is essential that each submodel within a model or model (federate) within a federation is tested individually and found to be sufficiently accurate before model interface testing begins. (Recall Principle 6 from Chapter 2!)

This form of testing deals with how well the submodels (or federates) are integrated with each other and is particularly useful for object-oriented and distributed simulations. Under the object-oriented paradigm, objects (a) are created with public and private interfaces, (b) interface with other objects through message passing, (c) are reused with their interfaces, and (d) inherit the interfaces and services of other objects.

Model interface testing assesses the accuracy of four types of interfaces, as identified by Sommerville (1996):

1. Parameter interfaces that pass data or function references from one object to another
2. Shared memory interfaces that enable objects to share a block of memory in which data are placed by one object and from which they are retrieved by other objects
3. Procedural interfaces that implement the concept of encapsulation under the object-oriented paradigm—an object provides a set of services (procedures) that can be used by other objects and hides (encapsulates) the way a service is provided from the outside world
4. Message-passing interfaces that enable an object to request the service of another object through message passing

Sommerville (1996) classifies interface errors into three categories:

1. Interface misuse occurs when an object calls another and incorrectly uses its interface. For objects with parameter interfaces, a parameter may be of the wrong type or may be passed in the wrong order, or the wrong number of parameters may be passed.
2. Interface misunderstanding occurs when object A calls object B without satisfying the underlying assumptions of object B's interface. For example, object A calls a binary search routine by passing an unordered list to be searched, when in fact the binary algorithm assumes that the list is already sorted.
3. Timing errors occur in real-time, parallel, and distributed simulations that use a shared memory or a message-passing interface.

User interface testing detects model representation errors created as a result of user-model interface errors or invalid assumptions about this interface. This form of testing is particularly important for testing human-in-the-loop and interactive simulations.

User interface testing assesses the interactions between the user and the model. The user interface is examined from low-level ergonomic aspects to instrumentation and controls and from human factors to global considerations of usability and appropriateness to identify potential errors (Miller *et al.,* 1995; Pressman, 1996; Schach, 1996).

4.1.3.15 Object-Flow Testing

Object-flow testing is similar to *transaction-flow testing* (Beizer, 1990) and *thread testing* (Sommerville, 1996). It assesses model accuracy by exploring the life cycle of an object during model execution. For example, a dynamic object (aircraft) can be marked for testing in the visual simulation environment for the model shown in Figure 4-2. Every time the dynamic object enters into a subroutine, the visualization of that subroutine is displayed. Every time the dynamic object interacts with another object within the subroutine, the interaction is highlighted. Examination of the way a dynamic object flows through the activities and processes and interacts with its environment during its lifetime in model execution is extremely useful for identifying errors in model behavior.

4.1.3.16 Partition Testing

Partition testing examines the model with the test data generated by analyzing the model's functional representations or partitions. It is accomplished by (a) decomposing both the model specification and its implementation into functional representations (partitions), (b) comparing the elements and prescribed functionality of each partition specification with the elements and actual functionality of the corresponding partition as it has been implemented in code, (c) deriving test data to test the functional behavior of each partition extensively, and (d) testing the model with the generated test data.

The model is decomposed into functional representations (partitions) through the use of symbolic evaluation techniques that maintain algebraic expressions of model elements and show model execution paths. These functional representations are the model computations. Two computations are equivalent if they are defined for the same subset of the input domain that causes a set of model paths to be executed and if the result of the computations is the same for each element within the subset of the input domain (Howden, 1976). Standard proof techniques show equivalence over a domain. When equivalence cannot be shown, partition testing is performed to locate errors or, as Richardson and Clarke (1985, p. 1488) state, to "increase confidence in the equality of the computations due to the lack of error manifestation." By involving both the model's specification and its implementation, partition testing can provide more comprehensive test data coverage than other test data generation techniques.

4.1.3.17 Predictive Validation

Predictive validation requires past input and output data from the system being modeled. The model is driven by past system input data and its forecasts are compared with the corresponding past system output data to test the predictive ability of the model (Emshoff and Sisson, 1970). Test data from test and evaluation uses of M&S are one example of how this technique is often used. Predictive validation also can evolve into the Model-Test-Model methodology, which uses the test data to make subsequent improvements to the model.

4.1.3.18 Product Testing

Product testing is conducted by the model or simulation developer after all submodels are successfully integrated (as demonstrated by the interface testing) and before the acceptance testing is performed by the model or simulation application sponsor or proponent. No contractor wants the product (model) to fail the acceptance test. Product testing serves to prepare for the acceptance testing. As such, the developer's quality control group must test the product and make sure that all requirements specified in the legal contract are satisfied before delivering the model to the model or simulation application sponsor (Schach, 1996).

As dictated by Principle 6 in Chapter 2, successfully testing each submodel or federate does not imply overall model or federation credibility. Interface testing and product testing are two techniques that must be performed to substantiate overall model credibility.

4.1.3.19 Regression Testing

Regression testing investigates the relationships between variables. In particular, it ensures that correcting errors and making changes in the model do not create other errors and adverse side effects. Usually the modified model is retested with the test data sets used previously. Successful regression testing requires the retention and management of old test data sets throughout the model development life cycle.

4.1.3.20 Sensitivity Analysis

Sensitivity analysis is performed by systematically changing the values of model input variables and parameters over some range of interest and observing the effect upon model behavior (Shannon, 1975). Unexpected effects may reveal invalidity. The input values also can be changed to induce errors to determine the sensitivity of model behavior to such errors. Sensitivity analysis can identify those input variables and parameters to which model behavior is very sensitive. Model validity then can be enhanced by ensuring that those values are specified with sufficient accuracy (Hermann, 1967; Miller, 1974a,b; Van Horn, 1971).

4.1.3.21 Special Input Testing

Special input testing consists of eight types of tests: boundary value, equivalence partitioning, extreme input, invalid input, real-time input, self-driven input, stress, and trace-driven input techniques. These techniques assess model accuracy by subjecting the model to a variety of inputs.

Boundary value testing examines the model's accuracy by using test cases on the boundaries of input equivalence classes. A model's input domain usually can be divided into classes of input data (known as equivalence classes) that cause the model to function the same way. For example, a traffic intersection model might specify the probability of left turn in a three-way turning lane as 0.2, the probability of right turn as 0.35, and the probability of traveling straight as 0.45. This probabilistic branching can be implemented by using a uniform random-number generator that produces numbers in the range $0 \leq rn \leq 1$. Thus, three equivalence classes are identified: $0 \leq rn \leq 0.2$, $0.2 < rn \leq 0.55$, and $0.55 < rn \leq 1$. Each test case from within a given equivalence class has the same effect on the model behavior, i.e., produces the same direction of turn.

In boundary analysis, test cases are generated just within, on top of, and outside of the equivalence classes (Myers, 1979). In the example above, the following test cases are selected for the left turn: 0.0, ±0.000001, 0.199999, 0.2, and 0.200001. In addition to generating test data on the basis of input equivalence classes, it also is useful to generate test data that will cause the model to produce values on the boundaries of *output* equivalence classes (Myers, 1979). The underlying rationale for this technique as a whole is that the most error-prone test cases lie along the boundaries (Ould and Unwin, 1986). Notice that invalid test cases used in the example will cause the model execution to fail; however, this failure should be as expected and meaningfully documented.

Equivalence partitioning testing partitions the model input domain into equivalence classes in such a manner that a test of a representative value from a class is assumed to be a test of all values in that class (Miller *et al.*, 1995; Perry, 1995; Pressman, 1996; Sommerville, 1996).

Extreme input testing is conducted by running the model or simulation with only minimum values, maximum values, or an arbitrary mixture of minimum and maximum values for the model input variables. For example, this technique allows the model user to test a proposed weapon system against extreme conditions that may not be obtainable in actual system testing.

Invalid input testing is performed by running the model or simulation under incorrect input data to determine whether the model behaves as expected. Unexplained behavior may reveal errors in model representations.

Real-time input testing is particularly important for assessing the accuracy of simulations built to represent embedded real-time systems. For example, different design strategies of a real-time software system built to control the operations of a manufacturing system can be studied using M&S. The model that represents the software design can be tested by running it with real-time input data that can be collected from the existing manufacturing system. Using real-time input data collected from a real system is particularly important to capture the timing relationships and correlations between input data points.

Self-driven input testing is conducted by running the model or simulation under input data randomly sampled from probabilistic models representing random phenomena in a real or future system. A probability distribution (e.g., exponential, gamma, weibull) can be fit to collected data, or triangular and beta probability distributions can be used in the absence of data, to model random input conditions (Banks *et al.*, 1996; Law and Kelton, 1991). Then, using random variate generation techniques, random values can be sampled from the probabilistic models to test the model validity under a set of observed or speculated random input conditions.

Stress testing tests the model's validity under extreme workload conditions. This is usually accomplished by increasing the congestion in the model. For example, the model in Figure 4-2 can be stress tested by increasing the number of flights between two locations to an extremely high value. Such an increase in workload may create unexpected high congestion in the model. Under stress testing, the model may exhibit invalid behavior; however, such behavior should be as expected and meaningfully documented (Dunn, 1987; Myers, 1979).

Trace-driven input testing is conducted by running the model or simulation under input trace data collected from a real system. For example, a system can be instrumented with monitors that collect data by tracing all system events. The raw trace data then are refined to produce the real input data for testing the model or simulation.

4.1.3.22 Statistical Techniques

Much research has been conducted in applying statistical techniques to model validation. Table 4-1 presents the statistical techniques proposed for model validation and lists related references.

Table 4-1. Statistical Techniques Proposed for Validation

Technique	*References*
Analysis of Variance	Naylor and Finger, 1967
Confidence Intervals/Regions	Balci and Sargent, 1984; Law and Kelton, 1991; Shannon, 1975
Factor Analysis	Cohen and Cyert, 1961
Hotelling's T^2 Tests	Balci and Sargent, 1981, 1982a, 1982b, 1983; Shannon, 1975
Multivariate Analysis of Variance —Standard MANOVA —Permutation Methods —Nonparametric Ranking Methods	Garratt, 1974
Nonparametric Goodness-of-Fit Tests —Kolmogorov-Smirnov Test —Cramer-Von Mises Test —Chi-square Test	Gafarian and Walsh, 1969; Naylor and Finger, 1967
Nonparametric Tests of Means —Mann-Whitney-Wilcoxon Test —Analysis of Paired Observations	Shannon, 1975
Regression Analysis	Aigner, 1972; Cohen and Cyert, 1961; Howrey and Kelejian, 1969
Theil's Inequality Coefficient	Kheir and Holmes, 1978; Rowland and Holmes, 1978; Theil, 1961
Time Series Analysis —Spectral Analysis —Correlation Analysis —Error Analysis	Fishman and Kiviat, 1967; Gallant et al., 1974; Howrey and Kelejian, 1969; Hunt, 1970; Van Horn, 1971; Watts, 1969 Watts, 1969 Damborg and Fuller, 1976; Tytula, 1978
t-Test	Shannon, 1975; Teorey, 1975

The statistical techniques listed in Table 4-1 require the system being modeled to be completely observable, i.e., that all data required for model validation can be collected from the system. The model is validated by using the statistical techniques to compare the model output data with the corresponding system output data after the model is run with the same input data as the real system. Model and system outputs are compared using multivariate statistical techniques to capture the correlation among the output variables. A recommended validation procedure based on the use of simultaneous confidence intervals follows.

Example 4-1. A Validation Procedure Using Simultaneous Confidence Intervals.

The behavioral accuracy (validity) of a simulation with multiple outputs can be expressed in terms of the differences between the corresponding model and system output variables when the model is run with the same input data and operational conditions that drive the real system. The range of accuracy of the jth *model output variable can be represented by the* jth *confidence interval (c.i.) for the differences between the means of the* jth *model and system output variables. The simultaneous confidence intervals (s.c.i.) formed by these confidence intervals are called the model range of accuracy (m.r.a.) (Balci and Sargent, 1984).*

Assume that there are k *output variables from the model and* k *output variables from the system as shown in Figure 4.3. Let* $(\underline{\mu}^m)' = [\mu_1^m, \mu_2^m, \ldots, \mu_k^m]$ *and* $(\underline{\mu}^s)' = [\mu_1^s, \mu_2^s, \ldots, \mu_k^s]$ *be the* k *dimensional vectors of the population means of the model and system output variables, respectively. Basically, there are three approaches for constructing the s.c.i to express the m.r.a. for the mean behavior.*

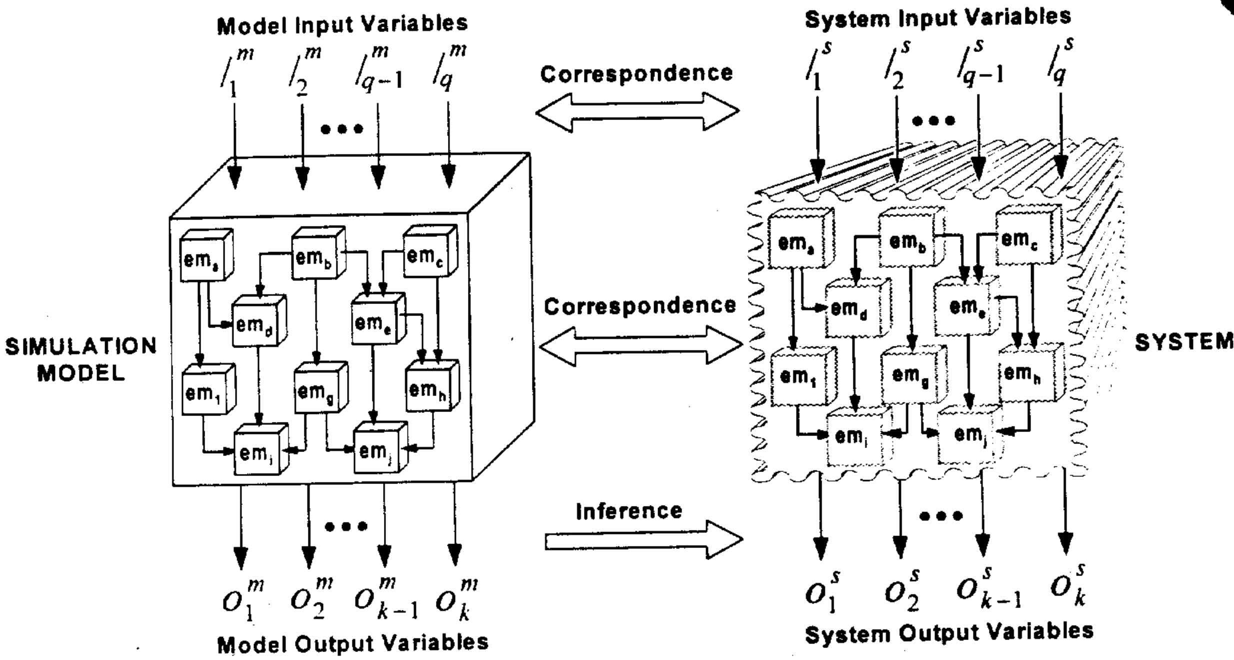

Figure 4-3. Model and System Characteristics

In Approach 1, the m.r.a. is determined by the 100(1−γ) % s.c.i. for $\underline{\mu}^m - \underline{\mu}^s$ *as*

$$[\underline{\delta} - \underline{\tau}] \tag{1}$$

where $\underline{\delta}' = [\delta_1, \delta_2, \ldots, \delta_k]$ *represents lower bounds and* $\underline{\tau}' = [\tau_1, \tau_2, \ldots, \tau_k]$ *represents upper bounds of the s.c.i. The modeler can be 100(1−γ) % confident that the true differences between the population means of the model and system output variables are simultaneously contained within (1).*

In Approach 2, the $100(1-\gamma^m)$ *% s.c.i. are first constructed for* $\underline{\mu}^m$ *as*

$$[\underline{\delta}^m, \underline{\tau}^m] \tag{2}$$

where $\left(\underline{\delta}^{m}\right)' = \left[\delta_1^m, \delta_2^m, \ldots, \delta_k^m\right]$ *and* $\left(\underline{\tau}^{m}\right)' = \left[\tau_1^m, \tau_2^m, \ldots, \tau_k^m\right]$. *Then, the* $100\left(1-\gamma^{s}\right)\%$ *s.c.i. are constructed for* $\underline{\mu}^{s}$ *as*

$$\left[\underline{\delta}^{s}, \underline{\tau}^{s}\right] \tag{3}$$

where $\left(\underline{\delta}^{s}\right)' = \left[\delta_1^s, \delta_2^s, \ldots, \delta_k^s\right]$ *and* $\left(\underline{\tau}^{s}\right)' = \left[\tau_1^s, \tau_2^s, \ldots, \tau_k^s\right]$. *Finally, using the Bonferroni inequality, the m.r.a. is determined by the following s.c.i. for* $\underline{\mu}^{m} - \underline{\mu}^{s}$ *with a confidence level of at least* $\left(1-\gamma^{m}-\gamma^{s}\right)$ *when the model and system outputs are dependent and with a level of at least* $\left(1-\gamma^{m}-\gamma^{s}+\gamma^{m}\gamma^{s}\right)$ *when the outputs are independent (Kleijnen, 1975):*

$$\left[\underline{\delta}^{m}-\underline{\tau}^{s}, \underline{\tau}^{m}-\underline{\delta}^{s}\right] \tag{4}$$

In Approach 3, the model and system output variables are observed in pairs and the m.r.a. is determined by the $100(1-\gamma)\%$ *s.c.i. for* $\underline{\mu}^{d}$, *the population means of the differences of paired observations, as*

$$\left[\underline{\delta}^{d}, \underline{\tau}^{d}\right] \tag{5}$$

where $\left(\underline{\delta}^{d}\right)' = \left[\delta_1^d, \delta_2^d, \ldots, \delta_k^d\right]$ *and* $\left(\underline{\tau}^{d}\right)' = \left[\tau_1^d, \tau_2^d, \ldots, \tau_k^d\right]$.

The m.r.a. is constructed with the observations derived from the model and system output variables by running the model with the same input data and operational conditions that drive the real system. If the simulation is self-driven, then the model input data come independently from the same populations or stochastic process as the system input data. Because the model and system input data are independent of each other, but come from the same populations, the model and system output data are expected to be independent and identically distributed. Hence, Approach 1 or 2 can be used. The use of Approach 3 in this case would be less efficient. If the simulation is trace-driven, the model input data are exactly the same as the system input data. In this case, the model and system output data are expected to be dependent and identical. Therefore, Approach 2 or 3 should be used.

Sometimes, the model or simulation application sponsor or proponent may specify an acceptable range of accuracy for a specific simulation. This specification can be made for the mean behavior of a stochastic simulation as

$$L \leq \underline{\mu}^{m} - \underline{\mu}^{s} \leq \underline{U}. \qquad (6)$$

where $\underline{L}' = [L_1, L_2, \ldots, L_k]$ *and* $\underline{U}' = [U_1, U_2, \ldots, U_k]$ *are the lower and upper bounds of the acceptable differences between the population means of the model and system output variables. In this case, the m.r.a. should be compared against Equation (6) to evaluate model validity.*

The shorter the lengths of the m.r.a., the more meaningful is the information they provide. The lengths can be decreased by increasing the sample sizes or by decreasing the confidence level. Such increases in sample sizes, however, may increase the cost of data collection. Thus, a trade-off analysis may be necessary among the sample sizes, confidence levels, half-length estimates of the m.r.a., data collection method, and cost of data collection. For details of performing the trade-off analysis, see Balci and Sargent, 1984.

4.1.3.23 Structural Testing

Structural testing (also called *white-box testing*) consists of six testing techniques: branch, condition, data flow, loop, path, and statement testing. Structural (white-box) testing evaluates the model based on its internal structure (how it is built), whereas functional (black-box) testing assesses the input-output transformation accuracy of the model. (Refer to Section 4.1.3.12.) Structural testing employs data flow and control flow diagrams to assess the accuracy of internal model structure by examining model elements such as statements, branches, conditions, loops, internal logic, internal data representations, submodel interfaces, and model execution paths.

Branch testing runs the model or simulation under test data to execute as many branch alternatives as possible, as many times as possible, and to substantiate their accurate operation. The more branches that test successfully, the more confidence is gained in the model's accurate execution with respect to its logical branches (Beizer, 1990).

Condition testing runs the model or simulation under test data to execute as many logical conditions as possible, as many times as possible, and to substantiate their accurate operation. The more logical conditions that test successfully, the more confidence is gained in the model's accurate execution with respect to its logical conditions.

Data flow testing uses the control flowgraph to explore sequences of events related to the status of data structures and to examine data-flow anomalies. For example, sufficient paths can be forced to execute under test data to ensure that every data element and structure is initialized before use or every declared data structure is used at least once in an executed path (Beizer, 1990).

Loop testing runs the model or simulation under test data to execute as many loop structures as possible, as many times as possible, and to substantiate their accurate operation. The more loop structures that test successfully, the more confidence is gained in the model's accurate execution with respect to its loop structures (Pressman, 1996).

Path testing runs the model or simulation under test data to execute as many control flow paths as possible, as many times as possible, and to substantiate their accurate operation. The more control flow paths that test successfully, the more confidence is gained in the model's accurate execution with respect to its control flow paths, but 100 percent path coverage is impossible to achieve for a reasonably large M&S application (Beizer, 1990).

Path testing is performed in three steps (Howden, 1976). In Step 1, the model control structure is determined and represented in a control flow diagram. In Step 2, test data is generated to cause selected model logical paths to be executed. Symbolic evaluation (Section 4.1.2.8) can be used to identify and classify input data based on the symbolic representation of the model. The test data is generated in such a way as to (a) cover all statements in the path, (b) encounter all nodes in the path, (c) cover all branches from a node in the path, (d) achieve all decision combinations at each branch point in the path, and (e) traverse all paths (Prather and Myers, 1987). In Step 3, by using the generated test data, the model is forced to proceed through each path in its execution structure, thereby providing comprehensive testing.

In practice, only a subset of all possible model paths is selected for testing due to budgetary constraints. Recent work has sought to increase the amount of coverage per test case or to improve the effectiveness of the testing by selecting the most critical areas to test. (Savvy readers may note that this technique is similar to the larger concept of VV&A tailoring that was addressed in Chapters 1 and 3.) The path prefix strategy is an adaptive strategy that uses previously tested paths as a guide in the selection of subsequent test paths. Prather and Myers (1987) prove that the path prefix strategy achieves total branch coverage.

The identification of essential paths is a strategy that reduces the path coverage required by nearly 40 percent (Chusho, 1987) by eliminating nonessential paths. Paths overlapped by other paths are nonessential. The model control flow graph is transformed into a directed graph whose arcs (called *primitive arcs*) correspond to the essential paths of the model. Nonessential arcs are called *inheritor arcs* because they inherit information from the primitive arcs. The graph produced during the transformation is called an *inheritor-reduced graph*. Chusho (1987) presents algorithms for efficiently identifying nonessential paths, reducing the control graph into an inheritor-reduced graph, and applying the concept of essential paths to the selection of effective test data.

Statement testing runs the model or simulation under test data to execute as many statements as possible, as many times as possible, and to substantiate their accurate operation. The more statements that test successfully, the more confidence is gained in the model's accurate execution with respect to its statements (Beizer, 1990).

4.1.3.24 Submodel/Module Testing

Submodel testing requires a top-down decomposition of the model into submodels. The executable model is instrumented to collect data on all input and output variables of a submodel. The system is instrumented (if possible) to collect similar data. Then, the behavior of each submodel is compared with the corresponding subsystem's behavior to judge the submodel's validity. If a subsystem can be modeled analytically, its exact solution can be compared against the simulation solution to assess its validity quantitatively.

As enumerated in Principle 6 in Chapter 2, validating each submodel individually does not imply sufficient validity for the whole model. Each submodel is found sufficiently valid with some allowable error. The allowable errors can accumulate to make the whole model invalid. Therefore, after each submodel is validated, the whole model itself must be tested.

4.1.3.25 Symbolic Debugging

This technique employs a debugging tool that allows the modeler to manipulate model execution while viewing the model at the source code level. By setting *breakpoints*, the modeler can interact with the entire model one step at a time, at predetermined locations, or under specified conditions. While using a symbolic debugger, the modeler may alter model data values or replay a portion of the model, i.e., execute it again under the same conditions. Typically, the modeler utilizes the information gathered with execution testing techniques (see Section 4.1.3.9) to isolate a problem or its proximity. Then the debugger is employed to determine how and why the error occurs.

Current state-of-the-art debuggers can view the runtime code as it appears in the source listing, set *watch* variables to monitor data flow, examine complex data structures, and even communicate with asynchronous input/output channels. The use of symbolic debugging can reduce greatly the debugging effort while increasing its effectiveness. Symbolic debugging allows the modeler to locate errors and check numerous circumstances that lead to errors (Whitner and Balci, 1989).

4.1.3.26 Top-Down Testing

Top-down testing is used with top-down model development. In top-down development, model construction starts with the submodels at the highest level and culminates with the routines at the base level, i.e., the ones that cannot be decomposed further. As each submodel is completed, it is tested thoroughly. When submodels with the same parent have been developed and tested, the submodels are integrated and their integration is tested. This process is repeated until the whole model has been integrated and tested. The integration of completed submodels need not wait for all submodels at the same level to be completed. Submodel integration and testing can be, and often is, performed incrementally (Sommerville, 1996).

Top-down testing begins with a test of the global model at its highest level. When testing a given level, calls to submodels at lower levels are simulated using *stubs*. A stub is a dummy submodel that has no function other than to let its caller complete the call. Fairley (1976) lists the following advantages of top-down testing: (a) model integration testing is minimized; (b) a working model is produced earlier in the development process; (c) higher level interfaces are tested first; (d) a natural environment for testing lower levels is created; and (e) errors are localized to new submodels and interfaces.

Some of the disadvantages of top-down testing are (a) thorough submodel testing is discouraged, because the entire model must be executed to perform testing; (b) testing can be expensive, because the whole model must be executed for each test; (c) adequate input data is difficult to obtain because of the complexity of the data paths and control predicates; and (d) integration testing is hampered because of the size and complexity of testing the whole model (Fairley, 1976).

4.1.3.27 Visualization/Animation

Visualization and animation of a simulation greatly assist in model V&V (Sargent, 1992). Displaying graphical images of internal (e.g., how customers are served by a cashier) and external (e.g., utilization of the cashier) dynamic behavior of a model during execution exhibits errors. For example, in visual simulation of a traffic intersection, the modeler can observe the arrival of vehicles in different lanes and their movements through the intersection as the traffic light changes. Visualizing the model as it executes and comparing it with the real traffic intersection can help identify discrepancies between the model and the system.

Seeing the model in action is very useful for uncovering errors; however, it does not guarantee model correctness (Paul, 1989). Therefore, visualization should be used with caution.

4.1.4 Formal V&V Techniques

Formal V&V techniques are based on formal mathematical proofs of correctness. If attainable, a formal proof of correctness is the most effective means of model V&V. Unfortunately, *if attainable* is the sticking point. Current formal proof of correctness techniques cannot be applied to even a reasonably complex M&S application; however, formal techniques serve as the foundation for other V&V techniques. The most commonly known eight techniques are briefly described below: (a) induction, (b) inference, (b) logical deduction, (d) inductive assertions, (e) lambda-calculus, (f) predicate calculus, (g) predicate transformation, and (h) proof of correctness (Khanna, 1991; Whitner and Balci, 1989).

Induction, inference, and **logical deduction** are simply acts of justifying conclusions on the basis of premises given. An argument is valid if the steps used to progress from the premises to the conclusion conform to established *rules of inference*. Inductive reasoning is based on invariant properties of a set of observations; assertions are invariants because their value is defined to be true. Given that the initial model assertion is correct, it stands to reason that if each path progressing from that assertion is correct and each path subsequently progressing from the previous assertion is correct, then the model must be correct if it terminates. Birta and Ozmizrak (1996) present a knowledge-based approach for M&S validation that uses a validation knowledge base containing rules of inference.

Inductive assertions assess model correctness based on an approach that is very close to formal proof of model correctness. It is conducted in three steps. In Step 1, input-to-output relations for all model variables are identified. In Step 2, these relations are converted into assertion statements and are placed along the model execution paths so that an assertion statement lies at the beginning and end of each model execution path. In Step 3, verification is achieved by proving for each path that, if the assertion at the beginning of the path is true and all statements along the path are executed, then the assertion at the end of the path is true. If all paths plus model termination can be proved, by induction, the model is proved to be correct (Manna *et al.*, 1973; Reynolds and Yeh, 1976).

Lambda Calculus (Barendregt, 1981) is a system that transforms the model into formal expressions by rewriting strings. The model itself can be considered a large string. Lambda calculus specifies rules for rewriting strings to transform the model into lambda calculus expressions. Using lambda calculus, the modeler can express the model formally to apply mathematical proof of correctness techniques to it.

Predicate calculus provides rules for manipulating predicates. A predicate is a combination of simple relations, such as *completed_jobs > steady_state_length*. A predicate will be either true or false. The model can be defined in terms of predicates and manipulated using the rules of predicate calculus. Predicate calculus forms the basis of all formal specification languages (Backhouse, 1986).

Predicate transformation (Dijkstra, 1975; Yeh, 1977) verifies model correctness by formally defining the semantics of the model with a mapping that transforms model output states to all possible model input states. This representation is the basis from which model correctness is proved.

Formal **proof of correctness** expresses the model in a precise notation and then mathematically proves that the executed model terminates and satisfies the requirements with sufficient accuracy (Backhouse, 1986; Schach, 1996). Attaining proof of correctness in a realistic sense is not possible under the current state of the art. The advantage of realizing proof of correctness is so great, however, that, when the capability is realized, it will revolutionize V&V.

4.2 Guidelines for Using the V&V Techniques

It is very important to understand the twelve principles of VV&A presented in Chapter 2 when applying the techniques just described to the VV&A process presented in Chapter 3. The principles help researchers, practitioners, and managers better understand M&S VV&A. They provide the underpinnings for the V&V techniques. Understanding and applying the principles is crucially important for the success of an M&S application.

Recall that, as stated in Principle 2 of Chapter 2, V&V is not a phase or step in the M&S life cycle but a continuous activity throughout the entire M&S life cycle. Table 4-2 shows the techniques that apply to the major stages of the generic VV&A process:

- Problem Definition
- M&S Approach
- M&S Requirements
- Conceptual Model
- M&S Design
- M&S Implementation
- M&S Application
- M&S Acceptability Assessment

The rows of Table 4-2 list the 76 V&V techniques described in this chapter, including a placeholder for the 18 statistical techniques shown in Table 4-1. These statistical techniques can be used to perform model validation quantitatively if data can be collected on the input and output processes of the system.

Table 4-2. Applicability of the V&V Techniques Throughout the M&S Life Cycle

	Formul. Problem	*M&S Approach*	*M&S Reqs.*	*Concep. Model*	*M&S Design*	*M&S Implem.*	*M&S Applic.*	*M&S Accept. Assess.*
Acceptance Testing							◆	◆
Alpha Testing						◆	◆	
Assertion Checking						◆		
Audit	◆	◆	◆	◆	◆	◆		
Authorization Testing						◆	◆	◆
Beta Testing						◆	◆	
Bottom-Up Testing						◆		
Boundary Value Testing						◆		
Branch Testing						◆		
Calling Structure Analysis				◆	◆	◆		
Cause-Effect Graphing	◆		◆	◆	◆	◆		
Comparison Testing		◆	◆	◆	◆	◆	◆	
Concurrent Process Analysis						◆	◆	
Condition Testing						◆		
Control Flow Analysis				◆	◆	◆		
Data Dependency Analysis	◆			◆	◆	◆		
Data Flow Analysis				◆	◆	◆		
Data Flow Testing						◆		
Data Interface Testing						◆	◆	
Debugging						◆		
Desk Checking	◆	◆	◆	◆	◆	◆		
Equivalence Partitioning Testing						◆		
Execution Monitoring						◆	◆	
Execution Profiling						◆	◆	
Execution Tracing						◆	◆	
Extreme Input Testing						◆		
Face Validation	◆	◆	◆	◆	◆	◆	◆	◆

Table 4-2. Applicability of the V&V Techniques Throughout the M&S Life Cycle (cont.)

	Formul. Problem	*M&S Approach*	*M&S Reqs.*	*Concep. Model*	*M&S Design*	*M&S Implem.*	*M&S Applic.*	*M&S Accept. Assess.*
Fault/Failure Analysis						◆	◆	
Fault/Failure Insertion Testing						◆	◆	
Field Testing							◆	
Functional Testing						◆	◆	
Graphical Comparisons						◆	◆	
Induction				◆	◆			
Inductive Assertions				◆	◆			
Inference				◆	◆			
Inspections	◆	◆	◆	◆	◆	◆	◆	◆
Invalid Input Testing						◆	◆	
Lambda Calculus				◆	◆			
Logical Deduction				◆	◆			
Loop Testing						◆		
Model Interface Analysis			◆	◆	◆			
Model Interface Testing						◆	◆	◆
Object-Flow Testing						◆	◆	◆
Partition Testing					◆	◆		
Path Testing						◆	◆	
Performance Testing							◆	◆
Predicate Calculus				◆	◆			
Predicate Transformation				◆	◆			
Predictive Validation						◆	◆	◆
Product Testing							◆	◆
Proof of Correctness				◆	◆			
Real-Time Input Testing						◆	◆	◆
Regression Testing						◆	◆	
Reviews	◆	◆	◆	◆	◆	◆	◆	◆
Security Testing							◆	◆

Table 4-2. Applicability of the V&V Techniques Throughout the M&S Life Cycle (cont.)

	Formul. Problem	*M&S Approach*	*M&S Reqs.*	*Concep. Model*	*M&S Design*	*M&S Implem.*	*M&S Applic.*	*M&S Accept. Assess.*
Self-Driven Input Testing						◆	◆	◆
Semantic Analysis					◆	◆		
Sensitivity Analysis						◆	◆	◆
Standards Testing							◆	◆
State Transition Analysis				◆	◆	◆		
Statement Testing						◆		
Statistical Techniques (Table 4-1)						◆	◆	◆
Stress Testing						◆	◆	◆
Structural Analysis				◆	◆			
Submodel/Module Testing						◆		
Symbolic Debugging						◆		
Symbolic Evaluation					◆			
Syntax Analysis						◆		
Top-Down Testing						◆		
Trace-Driven Input Testing						◆	◆	◆
Traceability Assessment				◆	◆	◆	◆	
Turing Test						◆	◆	◆
User Interface Analysis						◆	◆	◆
User Interface Testing						◆	◆	◆
Visualization/Animation						◆	◆	◆
Walkthroughs	◆	◆	◆	◆	◆	◆	◆	◆

Table 4-2 can be used to determine the V&V techniques that apply to each major stage of the M&S life cycle. From those applicable, the technique(s) for a particular V&V activity can be selected by considering the following: the model type as described in Figure 3-1; the problem to be solved through the use of M&S; the specific objectives of the M&S application; and the constraints of the application, including time, cost, and schedule.

The life cycle application of V&V is extremely important for successful completion of complex and large-scale M&S applications. How much to test or when to stop testing depends on the M&S application objectives. The V&V effort should continue until the modeler obtains sufficient confidence in the credibility and acceptability of the model or simulation results. *Sufficient confidence* is determined by the objectives of the M&S application.

Yet, it is recognized that applying V&V techniques throughout the life cycle is time-consuming and can be costly if not properly tailored to the relevant requirements of the problem. In practice, under pressure to complete an M&S application within a given timeframe, VV&A is usually sacrificed first. The sacrifice of VV&A means less than the delivery of a model without proven credibility and therefore without value to the decision maker. Remember the bottom line from Principle 2: Correction of errors early in development always costs less than correction of errors later. If you are worried about the cost of VV&A, it is better to spend a little up front than a lot later. During a meeting in the General's office or standing before a senior-level review board is not the time to realize that the sacrifice of VV&A was a mistake.

References

Ackerman, A.F., Fowler, P.J., & Ebenau, R.G. (1983). Software inspections and the industrial production of software. In Hans-Ludwig Hausen (Ed.), *Software validation: Inspection, testing, verification, alternatives*. Proceedings of the Symposium on Software Validation (pp. 13–40). Darmstadt, FRG.

Adrion, W.R., Branstad, M.A., & Cherniavsky, J.C. (1982). Validation, verification, and testing of computer software. *Computing Surveys, 14* (2), 159–192.

Aigner, D.J. (1972). A note on verification of computer simulation models. *Management Science, 18* (11), 615–619.

Allen, F.E. & Cocke, J. (1976). A program data flow analysis procedure. *Communications of the ACM, 19* (3), 137–147.

Backhouse, R.C. (1986). *Program construction and verification*. London: Prentice-Hall International (UK) Ltd.

Balci, O. (1988). The implementation of four conceptual frameworks for simulation modeling in high-level languages. In M.A. Abrams, P.L. Haigh, & J.C. Comfort (Eds.), *Proceedings of the 1988 Winter Simulation Conference* (pp. 287–295). Piscataway, NJ: IEEE.

Balci, O., Bertelrud, A.I., Esterbrook, C.M., & Nance, R.E. (1995). A picture-based object-oriented visual simulation environment. In C. Alexopoulos, K. Kang, W.R. Lilegdon, & D. Goldsman (Eds.), *Proceedings of the 1995 Winter Simulation Conference* (pp. 1333–1340). Piscataway, NJ: IEEE.

Balci, O. & Sargent, R.G. (1981). A methodology for cost-risk analysis in the statistical validation of simulation models. *Communications of the ACM, 24* (4), 190–197.

Balci, O. & Sargent, R.G. (1982a). Some examples of simulation model validation using hypothesis testing. In H.J. Highland, Y.W. Chao, & O.S. Madrigal (Eds.), *Proceedings of the 1982 Winter Simulation Conference* (pp. 620–620). Piscataway, NJ: North-Holland IEEE.

Balci, O. & Sargent, R.G. (1982b). Validation of multivariate response models using Hotelling's two-sample T^2 test. *Simulation, 39* (6), 185–192.

Balci, O. & Sargent, R.G. (1983). Validation of multivariate response trace-driven simulation models. In A.K. Agrawala & S.K. Tripathi (Eds.), *Performance '83* (pp. 309–323). North-Holland, Amsterdam.

Balci, O. & Sargent, R.G. (1984). Validation of simulation models via simultaneous confidence intervals. *American Journal of Mathematical and Management Sciences, 4* (3&4), 375–406.

Banks, J., Carson, J.S., & Nelson, B.L. (1996). *Discrete-event system simulation* (2nd ed.). Englewood Cliffs, NJ: Prentice-Hall.

Barendregt, H.P. (1981). *The lambda calculus: Its syntax and semantics*. New York: North-Holland.

Beizer, B. (1990). *Software testing techniques* (2nd ed.). New York: Van Nostrand Reinhold.

Birta, L.G. & Ozmizrak, F.N. (1996). A knowledge-based approach for the validation of simulation models: The foundation. *ACM Transactions on Modeling and Computer Simulation* (in press).

Chusho, T. (1987). Test data selection and quality estimation based on the concept of essential branches for path testing. *IEEE Transactions on Software Engineering, SE-13* (5), 509–517.

Cohen, K.J. & Cyert, R.M. (1961). Computer models in dynamic economics. *Quarterly Journal of Economics, 75* (1), 112–127.

Damborg, M.J. & Fuller, L.F. (1976). Model validation using time and frequency domain error measures (ERDA Report No. 76-152). Springfield, VA: NTIS.

Deutsch, M.S. (1982). *Software verification and validation: Realistic project approaches.* Englewood Cliffs, NJ: Prentice-Hall.

Dillon, L.K. (1990). Using symbolic execution for verification of Ada tasking programs. *ACM Transactions on Programming Languages and Systems, 12* (4), 643–669.

Dobbins, J.H. (1987). Inspections as an up-front quality technique. In G.G. Schulmeyer & J.I. McManus (Eds.), *Handbook of software quality assurance* (pp. 137–177). New York: Van Nostrand-Reinhold Company.

Dunn, R.H. (1984). *Software defect removal*, New York: McGraw-Hill.

Dunn, R.H. (1987). The quest for software reliability. In G.G. Schulmeyer & J.I. McManus (Eds.), *Handbook of software quality assurance* (pp. 342–384). New York: Van Nostrand-Reinhold Company.

Emshoff, J.R. & Sisson, R.L. (1970). *Design and use of computer simulation models.* New York: MacMillan.

Fairley, R.E. (1975). An experimental program-testing facility. *IEEE Transactions on Software Engineering, SE-1* (4), 350–357.

Fairley, R.E. (1976, July). Dynamic testing of simulation software. In *Proceedings of the 1976 Summer Computer Simulation Conference* (pp. 708–710). La Jolla, CA: Simulation Councils.

Fishman, G.S. &. Kiviat, P.J. (1967). The analysis of simulation generated time series. *Management Science, 13* (7), 525–557.

Forrester, J.W. (1961). *Industrial dynamics.* Cambridge, MA: MIT Press.

Fujimoto, R.M. (1990). Parallel discrete event simulation. *Communications of the ACM, 33* (10), 31–53.

Fujimoto, R.M. (1993). Parallel discrete event simulation: Will the field survive? *ORSA Journal on Computing, 5* (3), 213–230.

Gafarian, A.V. & Walsh, J.E. (1969). Statistical approach for validating simulation models by comparison with operational systems. In *Proceedings of the 4th*

International Conference on Operations Research (pp. 702–705). New York: John Wiley & Sons.

Gallant, A.R., Gerig, T.M., & Evans, J.W. (1974). Time series realizations obtained according to an experimental design. *Journal of the American Statistical Association, 69* (347), 639–645.

Garratt, M. (1974, July). Statistical validation of simulation models. In *Proceedings of the 1974 Summer Computer Simulation Conference* (pp. 915–926). La Jolla, CA: Simulation Councils.

Hermann, C.F. (1967). Validation problems in games and simulations with special reference to models of international politics. *Behavioral Science, 12* (3), 216–231.

Hollocker, C.P. (1987). The standardization of software reviews and audits. In G.G. Schulmeyer & J.I. McManus (Eds.), *Handbook of software quality assurance* (pp. 211–266). New York: Van Nostrand-Reinhold Company.

Howden, W.E. (1976). Reliability of the path analysis testing strategy. *IEEE Transactions on Software Engineering, SE-2* (3), 208–214.

Howden, W.E. (1980). Functional program testing. *IEEE Transactions on Software Engineering, SE-6* (2), 162–169.

Howrey, P. & Kelejian, H.H. (1969). Simulation versus analytical solutions. In T.H. Naylor (Ed.), *The design of computer simulation experiments* (pp. 207–231). Durham, NC: Duke University Press.

Hunt, A.W. (1970). Statistical evaluation and verification of digital simulation models through spectral analysis. Unpublished doctoral dissertation, University of Texas at Austin.

Khanna, S. (1991). Logic programming for software verification and testing. *The Computer Journal, 34* (4), 350–357.

Kheir, N.A. & Holmes, W.M. (1978). On validating simulation models of missile systems. *Simulation, 30* (4), 117–128.

King, J.C. (1976). Symbolic execution and program testing. *Communications of the ACM, 19* (7), 385–394.

Kleijnen, J.P.C. (1975). *Statistical techniques in simulation* (Vol. 2). New York: Marcel Dekker.

Knight, J.C. & Myers, E.A. (1993). An improved inspection technique. *Communications of the ACM, 36* (11), 51–61.

Law, A.M. & Kelton, W.D. (1991). *Simulation modeling and analysis* (2nd ed.). New York: McGraw-Hill.

Manna, Z., Ness, S., & Vuillemin, J. (1973). Inductive methods for proving properties of programs. *Communications of the ACM, 16* (8), 491–502.

Miller, D.K. (1975). Validation of computer simulations in the social sciences. In *Proceedings of the Sixth Annual Conference on Modeling and Simulation* (pp. 743–746). Pittsburg, PA.

Miller, D.R. (1974a, July). Model validation through sensitivity analysis. In *Proceedings of the 1974 Summer Computer Simulation Conference* (pp. 911-914). La Jolla, CA: Simulation Councils.

Miller, D.R. (1974b). Sensitivity analysis and validation of simulation models. *Journal of Theoretical Biology, 48* (2), 345-360.

Miller, L.A., Groundwater, E.H., Hayes, J.E., & Mirsky, S.M. (1995). Survey and assessment of conventional software verification and validation methods (Special Publication NUREG/CR-6316, Vol. 2). Washington, DC: U.S. Nuclear Regulatory Commission.

Myers, G.J. (1978). A controlled experiment in program testing and code walkthroughs/inspections. *Communications of the ACM, 21* (9), 760-768.

Myers, G.J. (1979). *The art of software testing.* New York: John Wiley & Sons.

Naylor, T.H. & Finger, J.M. (1967). Verification of computer simulation models. *Management Science, 14* (2), B92-B101.

Ould, M.A. & Unwin, C. (1986). *Testing in software development.* Great Britain: Cambridge University Press.

Page, E.H. & Nance, R.E. (1994, July). Parallel discrete event simulation: A modeling methodological perspective. In D.K. Arvind, R. Bagrodia, & J.Y-B. Lin (Eds.), *Proceedings of the Eighth Workshop in Parallel and Distributed Simulation* (PADS '94) (pp. 88-93). Los Alamitos, CA: IEEE Computer Society Press.

Paul, R.J. (1989). Visual simulation: Seeing is believing? In R. Sharda, B.L. Golden, E. Wasil, O. Balci, & W. Stewart (Eds.), *Impacts of recent computer advances on operations research* (pp. 422-432). New York: Elsevier.

Perry, W. (1995). *Effective methods for software testing.* New York: John Wiley & Sons.

Prather, R.E. & Myers, J.P., Jr. (1987). The path prefix software testing strategy. *IEEE Transactions on Software Engineering, SE-13* (7), 761-766.

Pressman, R.S. (1996). *Software engineering: A practitioner's approach* (4^{th} Ed.). New York: McGraw-Hill.

Ramamoorthy, C.V., Ho, S.F., & Chen, W.T. (1976). On the automated generation of program test data. *IEEE Transactions on Software Engineering, SE-2* (4), 293-300.

Rattray, C. (Ed.). (1990). *Specification and verification of concurrent systems.* New York: Springer-Verlag.

Reynolds, C. & Yeh, R.T. (1976). Induction as the basis for program verification. *IEEE Transactions on Software Engineering, SE-2* (4), 244-252.

Richardson, D.J. & Clarke, L.A. (1985). Partition analysis: A method combining testing and verification. *IEEE Transactions on Software Engineering, SE-11* (12), 1477-1490.

Rowland, J.R. & Holmes, W.M. (1978) Simulation validation with sparse random data. *Computers and Electrical Engineering, 5* (3), 37-49.

Sargent, R.G. (1992). Validation and verification of simulation models. In J.J. Swain, D. Goldsman, R.C. Crain, & J.R. Wilson (Eds.), *Proceedings of the 1992 Winter Simulation Conference* (pp. 104–114). Piscataway, NJ: IEEE.

Schach, S.R. (1996). *Software engineering* (3rd ed.). Homewood, IL: Irwin.

Schruben, L.W. (1980). Establishing the credibility of simulations. *Simulation, 34* (3), 101–105.

Shannon, R.E. (1975). *Systems simulation: The art and science.* Englewood Cliffs, NJ: Prentice-Hall.

Sommerville, I. (1996). *Software engineering* (5th ed.). Reading, MA: Addison-Wesley.

Teorey, T.J. (1975). Validation criteria for computer system simulations. *Simuletter, 6* (4), 9–20.

Theil, H. (1961). *Economic forecasts and policy.* Amsterdam, The Netherlands: North-Holland.

Turing, A.M. (1963). Computing machinery and intelligence. In E.A. Feigenbaum & J. Feldman (Eds.), *Computers and thought* (pp. 11–15). New York: McGraw-Hill.

Tytula, T.P. (1978, June). A method for validating missile system simulation models (Technical Report E-78-11). Redstone Arsenal, AL: U.S. Army Missile R&D Command.

Van Horn, R.L. (1971). Validation of simulation results. *Management Science, 17* (5), 247–258.

Watts, D. (1969). Time series analysis. In T.H. Taylor (Ed.), *The design of computer simulation experiments* (pp. 165–179). Durham, NC: Duke University Press.

Whitner, R.B. & Balci, O. (1989). Guidelines for selecting and using simulation model verification techniques. In E.A. MacNair, K.J. Musselman, & P. Heidelberger (Eds.), *Proceedings of the 1989 Winter Simulation Conference* (pp. 559–568). Piscataway, NJ: IEEE.

Wright, R.D. (1972). Validating dynamic models: An evaluation of tests of predictive power. In *Proceedings of the 1972 Summer Computer Simulation Conference* (pp. 1286–1296). La Jolla, CA: Simulation Councils.

Yourdon, E. (1985). *Structured walkthroughs* (3rd ed.). New York: Yourdon Press.

Yucesan, E. & Jacobson, S.H. (1992). Building correct simulation models is difficult. In J.J. Swain, D. Goldsman, R.C. Crain, & J.R. Wilson (Eds.), *Proceedings of the 1992 Winter Simulation Conference* (pp. 783–790). Piscataway, NJ: IEEE.

Yucesan, E. & Jacobson, S.H. (1996). Intractable structural issues in discrete event simulation: Special cases and heuristic approaches. *ACM Transactions on Modeling and Computer Simulation* (in press).

Additional Reading

Balci, O. (1986). Requirements for model development environments. *Computers & Operations Research, 13* (1), 53-67.

Balci, O. & Nance, R.E. (1987). Simulation model development environments: A research prototype. *Journal of Operational Research Society, 38* (8), 753-763.

Derrick, E.J. & Balci, O. (1995). A visual simulation support environment based on the DOMINO conceptual framework. *Journal of Systems and Software, 31* (3), 215-237.

Dijkstra, E.W. (1975). Guarded commands, non-determinacy and a calculus for the derivation of programs. *Communications of the ACM, 18* (8), 453-457.

Moose, R.L. & Nance, R.E. (1989). The design and development of an analyzer for discrete event model specifications. In R. Sharda, B.L. Golden, E. Wasil, O. Balci, & W. Stewart (Eds.), *Impacts of recent computer advances on operations research* (pp. 407-421). New York: Elsevier.

Nance, R.E. & Overstreet, C.M. (1987). Diagnostic assistance using digraph representations of discrete event simulation model specifications. *Transactions of the SCS, 4* (1), 33-57.

Overstreet, C.M. & Nance, R.E. (1985). A specification language to assist in analysis of discrete event simulation models. *Communications of the ACM, 28* (2), 190-201.

Stucki, L.G. (1977). New directions in automated tools for improving software quality. In R. Yeh (Ed.), *Current trends in programming methodology*, Vol. 2 (pp. 80-111). Englewood Cliffs, NJ: Prentice-Hall.

Yeh, R.T. (1977). Verification of programs by predicate transformation. In *Current Trends in Programming Methodology*, Vol. 2 (pp. 228-247). Englewood Cliffs, NJ: Prentice-Hall.

Chapter 5 — Accreditation

5.1 Definition and Background

Accreditation occurs at a key point in the process to solve a given problem. At this point, the person responsible for accepting the solution determines the model or simulation is sufficient for its intended use. Accreditation is a *decision*—a decision to use a model or simulation for a specific application (i.e., project or program). In fact, any time anyone uses a model to solve even a small, informal problem, a *de facto,* implicit decision (accreditation) is made. For formal programs, however, this decision is explicit. The decision is supported by as much information as is necessary to be credible. According to DoD Directive 5000.59, accreditation is "the official certification that a model or simulation is acceptable for a specific purpose."

Accreditation, then, must be associated with a specific purpose or application. This is what should be meant when someone asks if a model is accredited. At times, the term is used more broadly to cover other activities similar to accreditation. For example, a *class accreditation* is a determination that a model or simulation can apply to a class of applications (e.g., battalion-level armor operations). In this accreditation, a model or simulation is reviewed to determine its overall capabilities to model a segment of the battlespace. Even with a class accreditation, however, an accreditation must be performed when a specific application is defined for the model's or simulation's use. Another name for class accreditation is *capabilities assessment.*

In addition to the accreditation to use a model for a specific application, many decision makers also will examine the credibility of a model's or simulation's results, a process referred to as *results accreditation*. Results accreditation is usually done by both Modeling and Simulation (M&S) and subject-matter experts who review the results to determine their correctness.

At times, the overall application for the model or simulation will be critical and will have high visibility. In this instance, levels of management above the primary model or simulation user may make additional accreditations. These multiple accreditations give assurance to those higher levels of management that the model or simulation to be used is appropriate.

Because other activities are associated with accreditation, the best answer to the earlier question about model accreditation may be "What do you mean by *accredited*?"

The remainder of this chapter uses the term *accreditation* in its basic sense: the decision to use a model or simulation for a specific application. The next sections discuss the role of accreditation in the overall application process, the process that is used to support an accreditation, and the participants in accreditation and their responsibilities.

5.2 Accreditation's Role in the Overall Application Process

The overall application process is shown in Figure 5-1. As indicated in Chapter 1, Section 1.3, and Chapter 3, Section 3.3.1, the problem statement (or program product) drives the requirements and the selection of the approach to solving the problem (or developing the product). This approach may include the use of M&S. If so, those problem requirements to be addressed by M&S drive the M&S approach. An initial step in the M&S approach is to determine the M&S capabilities needed to address the requirements appropriately. These capabilities are acceptability criteria to be applied at initial model or simulation selection as well as during the acceptability assessment.

The team that selects the model or simulation to be used screens the M&S candidates against an initial set of acceptability criteria. The screening process compares the capabilities needed against the documented functionality of each M&S candidate. Based on this screening process, the team selects a set of models or simulations that provides the best chance of satisfying the requirements of the problem or project. Note the use of the word *chance*. M&S, like any other tool or methodology, has a probability of not working correctly. Causes of failure include errors inherent in the model or simulation (none is perfect), inaccurate model or simulation documentation, and problem requirements or characteristics that become apparent after the beginning of the application process. To minimize the chance of inaccurate results, project delay, or failure, steps should be taken to enhance the credibility of (the degree of confidence in) the model or simulation selected for this application. These steps include Verification and Validation (V&V) of the model's or simulation's functions important to the application as well as assessment of the model's or simulation's general characteristics to ensure they can satisfy project needs. The savvy application sponsor will check at key points of the overall application process to ensure each of these steps has been carried out correctly before proceeding to the next step.

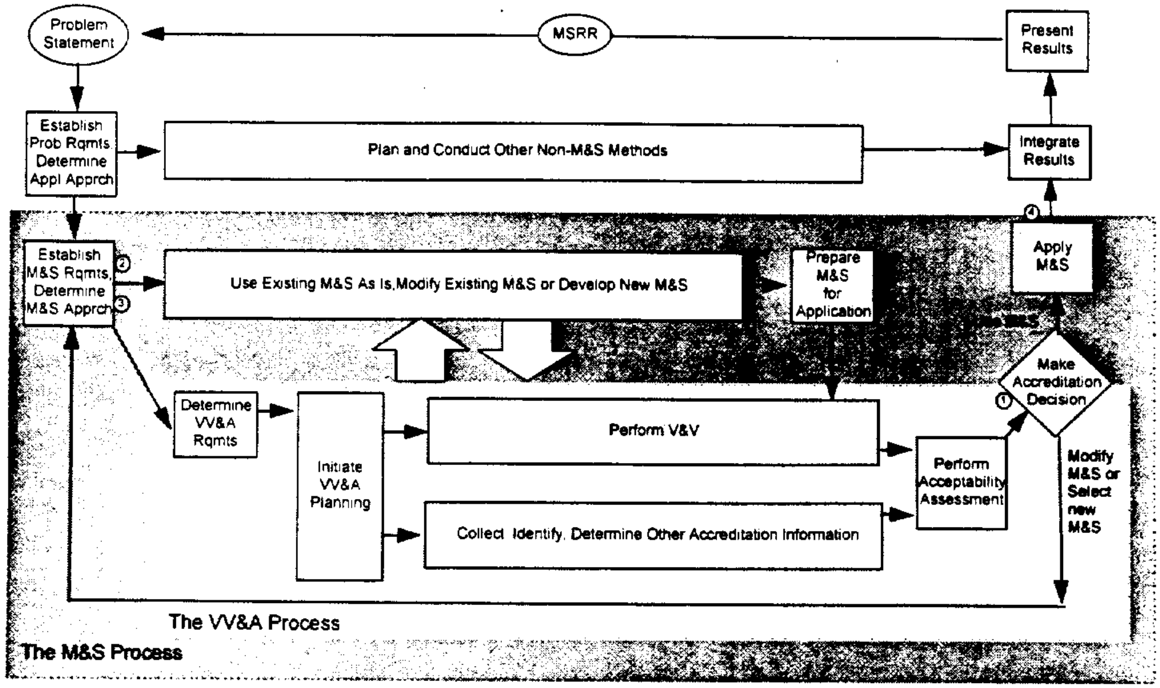

Figure 5-1. Verification, Validation, and Accreditation (VV&A) in the Application Process (Scheme of Things)

As a minimum, the sponsor should check when the acceptability assessment has been done and the recommendation for model or simulation use has been made, "circle" 1 in Figure 5-1. At this point, the accreditation authority decides to accept the model or simulation suite for use as is, to accept it for use with limitations, to use it only after enhancements have been made to the suite, to direct that additional V&V be performed, or not to use the model or simulation at all. Other points at which the accreditation authority or application sponsor may review and approve the work are when the M&S acceptability criteria have been developed ("circle" 2), when the initial model or simulation suite has been selected ("circle" 3), and when the M&S results have been generated ("circle" 4).

These checks give the application sponsor confidence that the M&S process will produce the results needed and that the M&S process is being carried out correctly. The greater the criticality of the overall application or the greater the inexperience with the model or simulation used, the more an application sponsor should review and approve critical process steps.

5.3 Process to Support Accreditation

This section describes the process leading up to and supporting accreditation. This process is shown in the boxes not shaded in Figure 5-2. Note that it encompasses the V&V process. For an application, V&V is a part of the accreditation process.

5.3.1 Accreditation Requirements

The accreditation process begins with the determination of accreditation requirements, based on the acceptability criteria developed in selecting the M&S approach. These requirements include the V&V requirements as well as other M&S characteristics needed and constraints based on application limitations. The process for determination of the V&V requirements is discussed in detail in Chapter 3. An overview of this process follows.

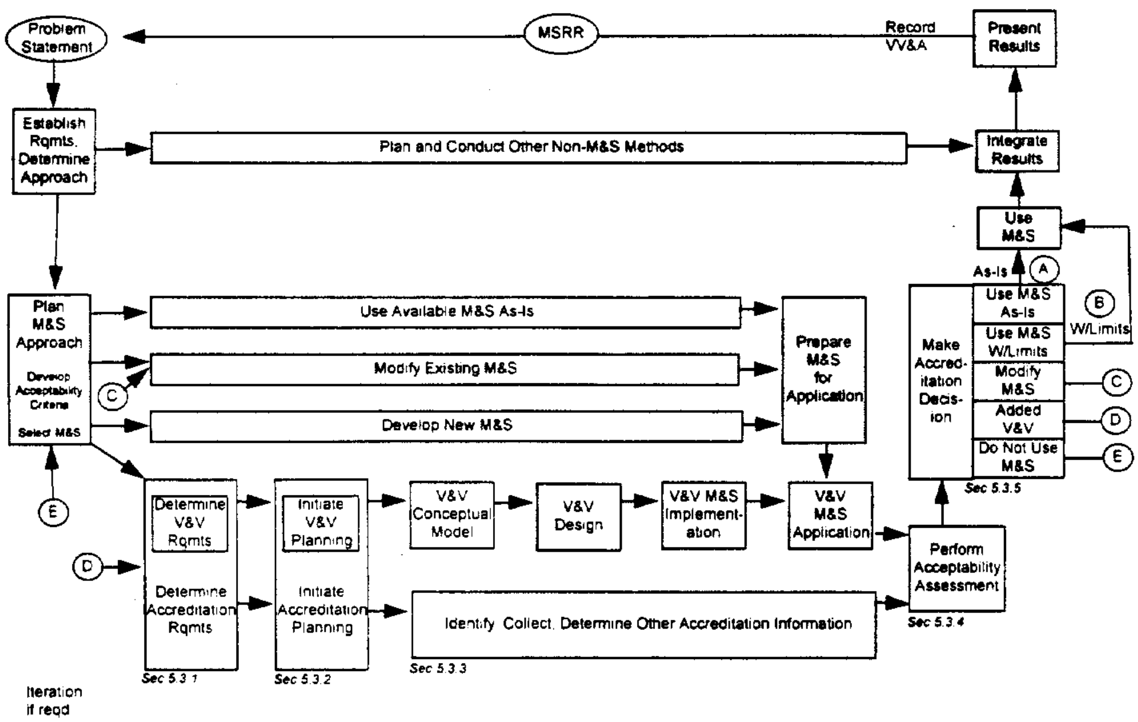

Figure 5-2. Process to Support Accreditation

The V&V requirements are determined first by defining the key M&S functions derived from the acceptability criteria. (These functions actually should have been determined as part of the earlier process to develop the acceptability criteria.) These key functions are prioritized in order of importance to the application. The V&V status of each of the key functions is then determined. The V&V status reflects whether V&V has been performed on this M&S function, the quality of the V&V performed, and the actual V&V findings. If the V&V status of a M&S function is sufficient for this application, no further V&V is required. If no V&V has been performed or the V&V accomplished is insufficient for this application, then a V&V requirement is generated. The sum of the V&V requirements for each key function makes up the initial V&V requirements for the application. The priority of the key function, and thereby the V&V requirement, can guide the V&V planner in determining the V&V that is to be accomplished, and potentially, in what sequence.

Other accreditation requirements include M&S characteristics that can affect the decision for the model's or simulation's approval and use. These factors include (a) model or simulation development and use history, (b) operational environment requirements, (c) configuration management status, (d) documentation status, and (e) other known capabilities and limitations of the model or simulation and supporting data bases.

An initial set of accreditation requirements, both V&V and non-V&V, is often used in the model or simulation selection process. For example, a model or simulation with a large set of V&V requirements is less likely to be selected over another model or simulation of similar capabilities with fewer V&V requirements.

The model or simulation development and use history is often a consideration for an accreditation authority in that an existing model or simulation with significant recent application use has more credibility than a new one with no history. The factors in development and use history are presented in Tables 5-1 and 5-2.

For many of these M&S development and use factors, the consideration is typically subjective and used in a comparative way. For example, one model or simulation may be a better choice than another because it was developed by an organization with extensive M&S development experience whereas the second was developed by an organization with little or no M&S development experience.

The M&S operational environment requirements are also a consideration for model or simulation selection and use because of the significant impact they can have on the resources required: facilities, time, and personnel. The factors in Table 5-3 are important for operational environment consideration.

Table 5-1. Developmental Factors to Consider

Development Factor	*Accreditation Consideration*
Initial model or simulation developers	Developer's reputation and SEI rating; M&S development experience
Development sponsor and reason for initial development (i.e., project, study)	Scope of sponsor's mission; scope of initial project or study requirements
M&S development methods applied	Good M&S development standards imply more efficient code and structure with fewer errors
Major M&S modifiers	Modifier's reputation and SEI rating; M&S modification experience
Modification sponsor	Scope of sponsor's mission
Reason for modification	Error correction or new capability added
M&S modification methods applied	Good M&S modification standards imply more efficient code/structure with fewer errors

For each major application, the factors in Table 5-2 apply.

Table 5-2. Use Factors to Consider

Use Factor	*Accreditation Consideration*
Major application description	Similarity of purpose and scope
Application sponsor	Scope of sponsor mission
Time frame of application	Currency of use
Critique of model or simulation use in application	Limitations discovered, operational problems, unexpected delays or problems, data base problems, overall success of model or simulation application

A third major factor to be considered for accreditation is the configuration management status of the model or simulation and its associated data bases. For a model or simulation to be usable by an application, it should be under competent configuration control. For the typical major DoD model or simulation, configuration management responsibility lies with the model's or simulation's proponent. Often, the sponsor leads a configuration control board with major model or simulation users as board members.

Table 5-3. Operational Environment Factors to Consider

Operational Environment Factor	*Accreditation Consideration*
Necessary hardware configuration needed to run the simulation including host type, processor speed, storage and storage devices, telecommunications links	Availability, cost, and scheduling of the necessary facility and configuration
Necessary software environment including operating system, language processors, data base systems, support software, display software	Availability and cost of obtaining, installing, or modifying software; availability and cost of personnel to make any software enhancements
Necessary personnel for model or simulation operation including number and experience level for model or simulation input data preparation, simulation execution and output analysis	Availability and cost of appropriate personnel including training
Necessary security including physical security of facility, data base security, personnel clearances	Cost of physical security; availability and cost of personnel with the appropriate clearances; time needed to obtain additional clearances

If configuration management has not been effective, a user cannot know what version of the model or simulation the application is using or what code, hardware, and data are really being used. Lack of configuration management may allow modifications to a model or simulation during an application without consideration of impact on overall operations.

Another major factor to be examined for accreditation is the model's or simulation's documentation. This factor relates to configuration management. Good configuration management usually implies good documentation. Poor or no configuration management leaves any M&S documentation suspect in terms of currency. The model's and simulation's documentation should have breadth (types of documentation, e.g., operator's manual, analyst's guide), depth (detail of documentation), accuracy, and currency (the model's or simulation's documentation matches the version being used).

A final major factor for accreditation is to review known limitations or problems with the model or simulation. A good configuration management system has such a list readily available. Other sources of this information are past or current users.

All these factors are possible considerations for the accreditation authority. Some or all of them may be appropriate for any specific application. Factors are selected to become accreditation requirements based on their perceived importance in making a credible accreditation decision as well as the estimated cost and time needed to gather the information. The appropriate Model and Simulation Resource Repository (MSRR) should have much of this information. As accreditation requirements are selected, they should be ranked, based on their priority to the application and on their importance to the accreditation authority.

5.3.2 Accreditation Planning

The application-specific accreditation requirements are satisfied based on the accreditation plan. The plan contains the list of requirements to be satisfied, the method of meeting each requirement, the agent responsible for each requirement, the overall resources needed, and the schedule for satisfying the requirements. A major subset of the accreditation plan is the V&V plan. Usually, this is a separate plan because it is the major work to be accomplished. It may be done by a group different from that satisfying all non-V&V requirements because of different skills or levels of expertise needed. The V&V planning process is discussed in more detail in Chapter 3.

Each requirement is examined, and the optimum method of requirement satisfaction is selected. The optimum is based on a trade-off of cost, resources, and time to complete. Each requirement satisfaction method then is grouped appropriately and integrated to give an overall approach to meeting the requirements. Requirements that drive the cost, resources, or schedule are re-examined to find more efficient ways of satisfying them. If no alternative can be found for a requirement that is excessively costly or time consuming, it should be reconsidered. Based on its priority, the requirement can be accepted as is, reformulated to make it easier to accomplish, or eliminated. Once the methods for all requirements are accepted, an integrated resource list and schedule is developed. If the V&V requirements are to be accomplished through a separate plan, they are documented separately. The approach to meeting all requirements is documented in the accreditation plan.

5.3.3 Accreditation Plan Execution

Once the accreditation plan has been approved, satisfaction of the requirements may begin. Chapter 3 provides a detailed description of the processes involved in V&V. The non-V&V requirements are met using the methods specified in the accreditation plan. These methods usually involve identifying sources of and collecting information, which should be documented. If execution of the accreditation plan is long or detailed, interim reports and reviews of progress may be appropriate.

5.3.4 Acceptability Assessment

The acceptability assessment reviews all accreditation information, both V&V and non-

V&V, and develops a list of capability voids, weaknesses, and mismatches of model or simulation functions and characteristics versus application acceptability criteria. The acceptability assessment team usually consists of the accreditation team and the V&V team, if it is separate. If modifications to the model or its data base are necessary to fill voids or correct weaknesses, approaches to these modifications along with the resources required and a schedule are developed and documented. If the voids or weaknesses can be avoided by limiting the uses of specific models or simulations, these limitations are documented. If there is a potential, yet undetermined weakness because of a lack of V&V, the additional V&V needed to determine if the weakness exists is estimated in terms of resources and time. The capability voids and weaknesses are analyzed together to develop an overall recommendation for model or simulation use, model or simulation use with limitations, model or simulation modifications, additional V&V, or model or simulation rejection. The results of the acceptability assessment and the recommendation with its rationale are documented in the acceptability assessment report and briefed to the accreditation authority.

5.3.5 Accreditation

The accreditation authority then has the responsibility to review the results of the acceptability assessment and, based on that information as well as other factors, make a decision. Among the other factors the accreditation authority may consider are a projected program schedule slip (for an acquisition program) or an anticipated budget decrease (or increase). The accreditation authority may ask the acceptability assessment team to develop additional information or different approaches to fill voids or eliminate weaknesses in a model's or simulation's capabilities before a decision is made. The decision can be one or a combination of the following:

(A) Use the model or simulation as it is for the application.
(B) Use the model or simulation with limitations in that use.
(C) Modify the model or simulation before use.
(D) Perform additional V&V.
(E) Do not use the model or simulation for this application.

Alternatives C through E incur additional costs and cause schedule changes. Alternative E is the most severe because it causes the process to begin again at developing the M&S approach.

The accreditation decision should be documented in a short report signed by the accreditation authority. At this point, the decision maker also should release the developed accreditation information to the MSRR to support future M&S applications.

5.3.6 Accreditation Process Tailoring

The process to support an accreditation decision is tailored to fit the needs of the accreditation authority or application sponsor and the application. For an application that is low in cost, with little national or DoD impact, or that produces results to be used in a low-level study, the credibility that the M&S tool used must possess is low. Hence, the accreditation requirement can be as simple as determining if the accreditation authority or application sponsor has used the M&S tool for a similar application. No VV&A planning is required, no acceptability assessment need be done, and the accreditation decision can be documented in a memorandum. On the other hand, for an acquisition program that has major, long-term budget implications and that will produce a significant new weapons system capability, the accreditation effort may use all the types of accreditation requirements described here, have a number of review and approval points, generate multiple interim reports, and have a large accreditation budget. Most applications fall somewhere between these extremes, and judgment will have to be used to assess the size of the accreditation effort correctly.

Other factors are considered in determining the size of the accreditation effort. For a given application, if a selected model or simulation has been recently and successfully used for a similar effort and the model's or simulation's configuration is well managed, then the results of the previous accreditation effort can be credibly relied on. A model or simulation well-established (documented) in the MSRR also makes information-gathering a relatively simple and easy task. For this reason, putting the basic model or simulation documentation, V&V information, and history of use in the MSRR is very important.

5.4 Roles

Any application has a number of key personnel roles. Table 5-4 summarizes these roles and responsibilities.

For some applications, some of these roles can be assumed by the same person. For example, the accreditation agent can also be the V&V agent. The number of people involved is a function of the size of the application and the amount of M&S to be applied.

Table 5-4. Personnel Requirements

Role	*Responsibility*
Accreditation Authority or Application Sponsor	Makes the accreditation decision; responsible for use of the M&S results and the overall application
Accreditation Agent	Manages the accreditation effort for a specific application; reports to the Accreditation Authority
V&V Agent	Manages the V&V effort for an application; reports to Accreditation Agent.
M&S Proponent	Responsible for development, modification, documentation, M&S configuration management, and V&V within a specific area of interest

5.5 Summary

The process leading to accreditation provides confidence to the application sponsor that the model or simulation can produce the results needed to develop the application's product. The magnitude of this process depends on the criticality of the application, the size of the M&S support for the application, and the amount of VV&A previously done for the selected model or simulation.

Chapter 6 — VV&A Common Reporting Formats

6.1 Introduction

The previous chapters provided an overview, principles, processes, and recommended procedures for Verification, Validation, and Accreditation (VV&A). This chapter offers formats for various VV&A reports. The following report formats are provided:

- *VV&A Acceptability Criteria Report*—This report documents the acceptability criteria for deciding if the model or simulation is suitable for the application. (See Table 6-1.)
- *Accreditation Plan*—This plan describes the information needed to approve the use of a model or simulation for a particular application and the planned approach to collect or develop that information. It also establishes the accreditation team and identifies the accreditation resources. (See Table 6-2.)
- *Verification and Validation (V&V) Plan*—This plan describes the V&V requirements, giving rationale, and the recommended V&V approach to satisfy those requirements. (See Table 6-3.)
- *Verification and Validation Report*—This report documents the results of executing the V&V plan. It provides data to the acceptability assessment. (See Table 6-4.)
- *Acceptability Assessment Report*—This report documents (a) the information needed to approve the use of a model or simulation for a particular application, (b) the information that was collected or developed based on the accreditation plan, (c) the comparison of the application M&S requirements to the model's or simulation's capabilities and limitations, (d) the model's or simulation's development and use history, (e) the model's or simulation's operating requirements and cost, (f) implications of the model's or simulation's limitations and constraints for use in this application, and (g) recommendations for changes to the model or simulation to use it for the application or to reduce application risk. (See Table 6-5.) The *Acceptability Assessment Report* is used in formulating the accreditation decision.
- *Accreditation Report*—This report documents the decision to use or not to use a model or simulation for a particular application. It may include limitations on a model's or simulation's use for this particular application. It also may contain direction for modification or for additional verification and validation to reduce overall application risk. The *Accreditation Report* provides the rationale for the decision. (See Table 6-6.)

The recommended report formats document average to large applications. Smaller applications may have less information in each report, and some of the reports may be combined, e.g., *V&V Plan* and *Accreditation Plan*. Larger applications may generate interim reports at the end of each V&V step. The actual report formats used should satisfy the needs of the application and should capture valuable VV&A effort for use in other applications. Automated aids for generating these reports will be available in the future to support report standardization and to reduce the cost and time of report preparation.

6.2 VV&A Reports in the Application Life Cycle

It is important for the M&S user to recognize the points in the overall application process at which reports, and specifically VV&A reports, are useful. Figure 6-1 gives an overview of the process for an application. The rectangular boxes are functions or steps in the process. The six square, shadowed boxes are reports. The figure outlines the entire application life cycle, but this guide covers only the VV&A portion. The shaded elements in Figure 6-1 are not covered by this manual.

The process starts with the overall application to be addressed. It establishes the basic problem requirements—what problems are to be considered, what answers or solutions are required, what the critical issues are, what the important characteristics and features of the application problems are, and so forth. Based on these application requirements, the approach to meeting them is developed. The application approach can combine several methods to satisfy application needs—field testing, laboratory testing, document research, or M&S. Note that M&S is but one method or tool. A separate approach is taken for non-M&S methods selected to satisfy a subset of the application requirements (subprocess at the top of Figure 6-1), but the results are integrated with the results from an M&S process.

The application requirements to be satisfied by M&S should be clearly identified. These M&S requirements will drive the development of an M&S approach. The M&S approach will direct the types and the combination of M&S that will be used to satisfy specific application requirements. Because application planners typically have multiple M&S candidates to satisfy any particular requirement, a trade-off assessment is made to determine the best M&S suite (or single model) for the application. This assessment may include the V&V status of a model or simulation or its previous accreditation and use. The assessment may suggest use of a particular model as it is (without modification), use of a particular model with some changes, or development of a new model. Once the M&S approach is selected, the VV&A process for these models or simulations may begin.

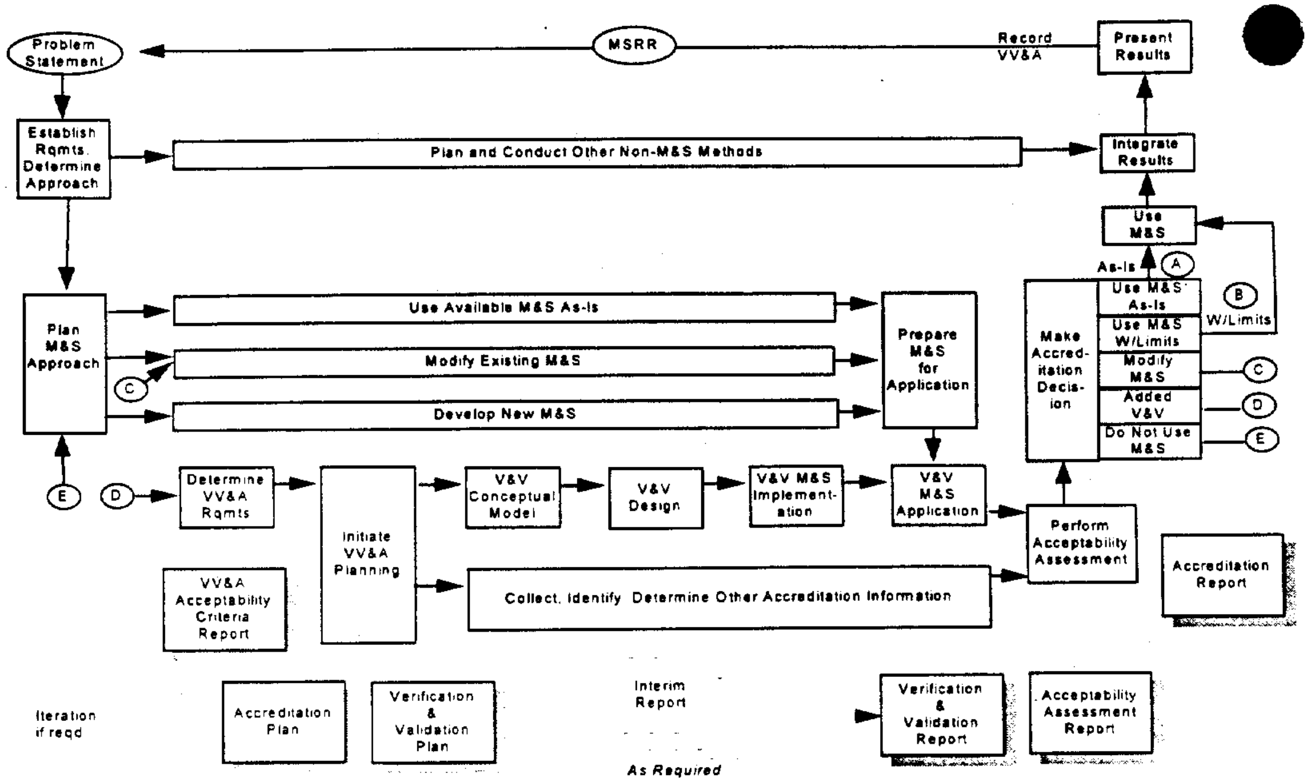

Figure 6-1. VV&A Reports in the M&S Application Life Cycle

The M&S requirements are used to derive VV&A acceptability criteria. These acceptability criteria specify not only the M&S functions and objects that are needed for the application but also the level of fidelity needed for each function and object. These acceptability criteria are documented in the *VV&A Acceptability Criteria Report*. (See Table 6-1.)

The VV&A planning begins with the development of the accreditation plan. It is driven by the application requirements to be met by the M&S approach selected. It uses the acceptability criteria from the *VV&A Acceptability Criteria Report*. The plan also identifies the other types of information (e.g., model or simulation history) that can be used to determine acceptability. This information is documented in the *Accreditation Plan*. (See Table 6-2.)

If an M&S suite is selected for the application, VV&A plans for each model or simulation should be selected. If the VV&A activity is extensive for several models or simulations, integrating the actions and schedules may be useful to eliminate competition for limited resources and to eliminate redundancy. This integration also can be documented in integrated VV&A reports.

Concurrent with the accreditation plan, the verification and validation plan can be developed and documented in the *V&V Plan*. (See Table 6-3.) The *V&V Plan* prioritizes V&V functions and objects based on the application requirements. It also will consider the V&V that has already been performed on the model or simulation by previous applications. When the *VV&A Plans* are complete and approved by the application manager, the V&V process can begin. The process for collecting non-V&V information needed for accreditation (shown in Table 6-2, sections F and G) may also begin. The V&V process is described in Chapter 3.

When the V&V is completed, it can be documented in the *V&V Report*. (See Table 6-4.) If the V&V activity warrants interim reports at the end of some or all of the V&V steps (e.g., conceptual model validation, design verification), they should be patterned after the applicable sections of the *V&V Report*. Interim reports are input to the final *V&V Report*.

The report format has a section for verification results and another section for validation results. Some of the V&V techniques can be considered for both verification and validation purposes (e.g., sensitivity analysis). Their results can be documented in each section, based on verification or validation criteria, or combined in a single section. With the approval of the application manager, the *V&V Report* can be sent to the Modeling and Simulation Resource Repository (MSRR) for archiving and use by future applications.

The acceptability assessment can be performed using the V&V results and non-V&V information collected. The acceptability assessment considers whether the model's or simulation's capabilities meet or exceed the application requirements. The risk, cost, schedule, and other implications of not meeting the requirements also are evaluated. Based on this assessment, recommendations are developed for review by the accreditation authority. The assessment process, the assessment results, and the recommendations, along with their rationales, are documented in the *Acceptability Assessment Report*. (See Table 6-5.) The *Acceptability Assessment Report*, with the approval of the accreditation authority, is sent to the MSRR.

Based on the *Acceptability Assessment Report* and other information and considerations, the accreditation decision is made. The decision and its rationale are documented in a short *Accreditation Report*. (See Table 6-6.) The decision can have any of the following outcomes:

(A) The model or simulation can be used as is for the application.

(B) The model or simulation can be used with certain limitations in application area. This decision may or may not necessitate a change to the M&S approach.

(C) Modifications to the model or simulation must be made to enhance its capability. This decision will require follow-up V&V.

(D) Additional V&V must be performed before the model or simulation can be accredited for this application.

(E) This model or simulation cannot be used for this application. This decision will necessitate changing the M&S approach, or perhaps even the application approach, and can have a significant impact on cost and schedule.

Table 6-1. *VV&A Acceptability Criteria Report* Format

A. **Application Description**—description of overall program for which accreditation will be accomplished

1. Program name
2. Short description
3. Program sponsor or responsible agency
4. Major program issues and objectives
5. Program importance and major risks
6. Program approach and methodology summary
7. Program schedule summary

B. **Application M&S Requirements and Acceptability Criteria**

1. Major M&S requirement areas (overview)
2. Requirement Area 1—Section B.2 is repeated for Requirement Area 2 through Requirement Area *N*.
    a. Major requirement area description
    b. Priority and importance of area to application accomplishment
    c. List of objects and functions with acceptability criteria—may include priority and importance of each object and function

ATTACHMENTS:

M&S Requirements document (if any)

Program Requirements document (if any)

Table 6-2. *Accreditation Plan* Format

A. Application Description and M&S Approach

1. Description of overall program for which accreditation will be accomplished
   a. Program name
   b. Short description
   c. Program sponsor or responsible agency
   d. Major program issues and objectives
   e. Program importance and major risks
   f. Program approach and methodology summary
   g. Program schedule summary
2. Program M&S methodology
   a. Model or simulation requirements (general)
   b. Model or simulation selected (or candidates)
   c. Proposed model's or simulation's use in decision process (integration with other methods and data)
3. Accreditation officials
   a. Accreditation authority
   b. Accreditation agent and team

B. Model Description

1. Model description
   a. Title
   b. Version
   c. Scope and overview
2. Model sponsor
3. Model configuration manager
4. Proposed use in decision process (integration with other methods and data)
5. Key objects and functions represented (see Section D.2 for complete list)
6. Operating environment (intended host hardware, software)
7. Key sources of data

C. Application M&S Requirements and Acceptability Criteria

1. Major M&S requirement areas (overview)
2. Requirement Area 1—Section C.2 is repeated for Requirement Area 2 through Requirement Area *N*.
   a. Major requirement area description
   b. Priority and importance of area to application accomplishment
   c. List of objects and functions with acceptability criteria—may include priority and importance of each object and function

D. Model Capability

1. Major model capability areas (overview)
2. List of model objects and functions represented
3. Comparison of model capability areas to application requirements areas—Will model be used in each application requirements area?
4. Major model limitations for each object and function

Table 6-2. *Accreditation Plan* Format (continued)

E. V&V Plan Summary

1. Verification Plan Summary
    a. Verification approach overview
    b. List of verification activities for each required area—For each model section in verification plan, provide the verification method and the verification agent.
2. Validation Plan Summary
    a. Validation approach overview
    b. List of validation activities for each required area—For each model section in validation plan, provide the validation method and the validation agent.
3. Data Verification, Validation, and Certification (VV&C) Plan Summary
    a. VV&C approach overview
    b. List of VV&C activities for each required area—For each data base section in VV&C plan, provide the VV&C method and the VV&C agent.
4. Schedule integrating all verification, validation, and data VV&C activities

F. Other Accreditation Information Requirements

1. Model or simulation development and use history
    a. Model development
        (1) Initial model developers and development sponsor
        (2) Reason for initial development (e.g., project, study)
        (3) Model development methods applied
        (4) Major model modifiers and modification sponsors
        (5) Reason for modifications (e.g., project, study)
        (6) Model modification methods applied
    b. Model or simulation use—For each major application, the following information is desired:
        (1) Major application and application sponsor
        (2) Time frame of application
        (3) Critique of model or simulation use in application, e.g., limitations discovered, operational problems, unexpected delays or costs, data base problems, overall success of model or simulation application
2. Implications of operational environment requirements
    a. Necessary hardware configuration needed to run the simulation including implications of storage and storage devices, processor speed, telecommunications links
    b. Necessary software environment including operating system, language processors, support software, display software, data base systems
    c. Necessary personnel for operation including number and expertise level for modeling and simulation operation and analysis
    d. Necessary security requirements
3. Description of configuration management system and process being applied to this model or simulation including listing of Configuration Control Board members/chair
4. Model or simulation documentation available including breadth (types of documentation), depth (detail of documentation), accuracy, and currency
5. Other known capabilities/limitations of the model or simulation or its data base

Table 6-2. *Accreditation Plan* Format (continued)

G. Plan to Collect Other Accreditation Information

1. Sources of information about the model or simulation
   a. Repositories
   b. Configuration manager
   c. Developer
   d. Users
   e. Project and study reports, including other VV&A reports
   f. Documentation
2. Schedule and resources for collecting the information—considerations include security, volume of documentation/information, organizational sensitivities

H. Accreditation Plan Integrated Schedule/Resources—an integrated schedule with resources planned for all V&V and accreditation information development and collection

ATTACHMENTS:
M&S Requirements document (if any)
M&S Selection Report (if any)

Table 6-3. *Verification and Validation Plan* Format

A. Application Description and M&S Approach

1. Description of overall program for which V&V is being accomplished
    a. Program name
    b. Short description
    c. Program sponsor or responsible agency
    d. Major program issues and objectives
    e. Program importance and major risks
    f. Program approach and methodology summary
    g. Program schedule summary
2. Program M&S methodology
    a. Model or simulation requirements (general)
    b. Model or simulation selected (or candidates)
    c. Proposed model's or simulation's use in decision process (integration with other methods and data)

B. Model Description

1. Model description
    a. Title
    b. Version
    c. Scope and overview
2. Model sponsor
3. Model configuration manager
4. Proposed use in decision process (integration with other methods/data)
5. Key objects and functions represented (see Section D.2 for complete list)
6. Operating environment (intended host hardware, software)
7. Key sources of data

C. Application M&S Requirements and Acceptability Criteria

1. Major M&S requirement areas (overview)
2. Requirement Area 1—Section C.2 is repeated for Requirement Area 2 through Requirement Area *N*.
    a. Major requirement area description
    b. List of objects and functions with acceptability criteria

D. Model Capability

1. Major model capability areas (overview)
2. List of model objects and functions represented
3. Comparison of model capability areas to application requirements areas—Will model be used in each application requirements area?
4. Major model limitations object and function

Table 6-3. *Verification and Validation Plan* Format (continued)

E. Model V&V Status

1. List of model objects and functions with verification status and validation status given
    a. Verification status is listed separately from validation status
    b. Each object and function status includes the following:
        (1) What specific effort provided V or V
        (2) When it was accomplished
        (3) What model version it was accomplished on
        (4) Pointer to detailed V&V report containing this specific information

F. Model V&V Requirements

1. List of model objects and functions with verification requirements and validation requirements
    a. Correlation to list of activities in Sections F.2 and F.3
    b. Importance or risk of not performing V or V
2. List of individual verification activities to be conducted
3. List of individual validation activities to be conducted

G. Verification Plan

1. Overview of all verification activities
2. Verification Activity 1—Section G.2 is repeated for Verification Activity 2 through Verification Activity *N*.
    a. Verification activity approach, which includes the following:
        (1) Model sections to be verified
        (2) Verification methods to be employed
        (3) Information and data sources
    b. Verification agents, key players
    c. Verification activity schedule (with milestones)
    d. Resources required
3. Integrated schedule and resources required layout for Verification Activity 1—Section G.3 is repeated for Verification Activity 2 through Verification Activity *N*.

H. Validation Plan

1. Overview of all validation activities
2. Validation Activity 1—Section H.2 is repeated for Validation Activity 2 through Validation Activity *N*.
    a. Validation activity approach, which includes the following:
        (1) Model sections to be validated
        (2) Validation methods to be employed
        (3) Information and data sources
    b. Validation agents, key players
    c. Validation activity schedule (with milestones)
    d. Resources required
3. Integrated schedule and resources required layout for Validation Activity 1—Section H.3 is repeated for Validation Activity 2 through Validation Activity *N*.

Table 6-3. *Verification and Validation Plan* Format (continued)

I. Data Verification, Validation and Certification (VV&C) Plan *(if required or separate from other V&V plans)*

1. Overview of all data VV&C activities
2. Data VV&C Activity 1—Section I.2 is repeated for VV&C Activity 2 through VV&C Activity *N.*
   a. Data VV&C activity approach, which includes the following:
      (1) Model data base sections needing VV&C
      (2) Data VV&C methods to be employed
      (3) Information and data sources
   b. Data VV&C agents, key players
   c. Data VV&C activity schedule (with milestones)
   d. Resources required
3. Integrated schedule and resources required layout for VV&C Activity 1—Section I.3 is repeated for VV&C Activity 2 through VV&C Activity *N.*

J. Integrated Verification and Validation

1. Schedule integrating all verification, validation, and data VV&C activities
2. Summary of resources for all verification, validation, and data VV&C activities

ATTACHMENTS:

M&S Requirements document (if any)
M&S Selection Report (if any)

Table 6-4. *Verification and Validation Report* Format

A. Executive Summary

1. Summary of V&V Plan
2. Summary of all sections of this report

B. Differences from V&V Plan

1. Verification differences—list of differences in the executed verification activity from the planned verification
2. Validation differences—list of differences in the executed validation activity from the planned validation
3. Verification, Validation, and Certification (VV&C) differences—list of differences in the executed VV&C activity from the planned VV&C

C. V&V Results

1. Verification results—Section C.1 is repeated for Verification Area 2 through Verification Area *N*.
   a. Verification Area 1 description
   b. Model section(s) verified
   c. Verification approach taken
   d. Schedule of activities, resources used
   e. Verification agent
   f. Verification results
2. Validation results—Section C.2 is repeated for Validation Area 2 through Validation Area *N*.
   a. Validation Area 1 description
   b. Model section(s) validated
   c. Validation approach taken
   d. Schedule of activities, resources used
   e. Validation agent
   f. Validation results
3. Data VV&C results—Section C.3 is repeated for VV&C Area 2 through VV&C Area *N*.
   a. Data VV&C Area 1 description
   b. Model section(s) needing data VV&C
   c. VV&C approach taken
   d. Schedule of activities, resources used
   e. Data VV&C agent
   f. Data VV&C results

D. V&V Summary—contains summary of V&V activities performed, the integrated schedule of performance, and a summary of resources used

ATTACHMENT:
Verification and Validation Plan

Table 6-5. *Acceptability Assessment Report* Format

A. Summary
1. Application description
2. M&S approach
3. Model description

B. Application M&S Requirements and Acceptability Criteria
1. Major M&S requirement areas (overview)
2. Major Requirement Area 1 description with acceptability criteria—Section B.2 is repeated for Requirement Area 2 through Requirement Area *N*.

C. Model Capability
1. Major model capability areas (overview)
2. List of model objects and functions represented
3. Comparison of model capability areas to application requirements areas—Will model be used in each application requirements area?
4. Major model limitations of each object and function

D. V&V Report Summary
1. Verification results summary
   a. Verification approach overview
   b. List of verification activities accomplished for each required area—For each verified model section, provide the verification method, the verification agent, and verification result.
2. Validation report summary
   a. Validation approach overview
   b. List of validation activities accomplished for each required area—For each validated model section, provide the validation method, the validation agent, application requirement, and model capability and accuracy.
3. Data Verification, Validation, and Certification (VV&C) report summary
   a. VV&C approach overview
   b. List of VV&C activities accomplished for each required area—For each data base section for which VV&C is needed, provide the VV&C method, the VV&C agent, and the VV&C result.

E. Comparison Analysis of Requirements versus Capabilities—For each major model section that had an application requirement, provide the following information:
1. Major model section name and short description
2. Application requirements for this section
3. Model capability/accuracy results of V&V activity
4. Comparison of requirements to model capability—includes analysis of differences and implications for application in terms of risk, cost, schedule

F. Comparison Analysis Summary
1. Prioritized list of model sections that do not meet application requirements
   a. Prioritized in terms of risk to the application
   b. Includes recommendations for reducing risk with cost and schedule implications of risk-reduction action

Table 6-6. *Accreditation Report* Format

A. Summary of Application and M&S Approach

1. Description of overall program for which accreditation applies
2. Program M&S methodology
   a. M&S requirements for selected model or simulation
   b. Model or simulation selected
   c. Model or simulation use in decision process (integration with other methods/data)
   d. Model or simulation description (title, version, overview)

B. Accreditation

1. Accreditation summary—decision for model or simulation useability for this application. It can be one of the following alternatives. It may also be a combination of b and c.
   a. The model or simulation will be used as described in M&S Requirements Plan for this application.
   b. The model or simulation will be used as described in M&S Requirements Plan for this application with limitations.
   c. The model or simulation will be used as described in M&S Requirements Plan for this application with modifications.
   d. The model or simulation requires additional V&V to be considered suitable for accreditation.
   e. The model or simulation will not be used for this application as described in M&S Requirements Plan.
2. ***If alternative b is selected.*** Limitations of model or simulation use for this application—the following is a list of limitations. For each limitation, include this information:
   a. Limitation description
   b. Rationale or risk involved in not imposing limitation
3. ***If alternative c or d is selected.*** Changes that will be made for the model to be used for this application—The following is the list of modifications to be made or additional V&V to be done. For each modification, include the following information:
   a. Modification enhancement or additional V&V description
   b. Rationale or risk involved in not making modification or additional V&V
4. ***If alternative e is selected.*** Follow-on accreditation actions—can include requirements for new annexes to be added to the V&V Plan and/or Accreditation Plan for the required modifications or additional V&V and may include the requirement for a supplemental accreditation decision and report

ATTACHMENT:
Acceptability Assessment Report