Part I

Introduction

Chapter **1**

Introduction

1.1 OPENING

This chapter serves as motivation for learning about systems Verification, Validation and Testing (VVT) as well as a map for using the book as a reference source on this complex and multifaceted process. We emphasize here the multitude of reasons for applying VVT. It sets the tone for the subject matter we hope to cover. It gives the reader insight into the attitudes of the author and the care with which the book was prepared. A clear statement is made of the purpose for which the book has been written.

The book is a compendium of facts about systems VVT. In fact, we think little has yet been published that is as comprehensive on this subject. By listing the potential audience for the book, we hope to encourage its wide distribution and to increase among engineers, managers, academicians and students an appreciation of the benefits of rigorously applying VVT to almost every endeavor involving a product or service, be it for purposes commercial, private or public. This chapter contains the following elements:

Opening. This part provides a background, purpose and the intended audience of the book. In addition, it describes its structure and contents as well as the scope of application and some terminology descriptions.

VVT systems and process. This part introduces VVT systems and processes as components of engineered systems. In addition, it describes basic VVT definitions and elaborates on the fundamental VVT dilemmas. Also, this part describes modeling of systems and VVT lifecycle as well as modeling of VVT processes and risks as cost and time drivers.

Verification, Validation, and Testing of Engineered Systems, Avner Engel

Canonical systems VVT paradigm. This part introduces the concept of canonical systems VVT paradigm which includes phases of systems' lifecycle, views of systems and VVT aspects of systems.

Methodology application. This part introduces methodology application including VVT methodology overview, VVT tailoring and typical VVT documentation.

1.1.1 Background

The manufacturing industry used to be concerned with the design, development, production and maintenance of stand-alone products, whether simple or complex. Today, however, manufacturing has broadened its scope to include products, services or solutions that include a variety of components, integrate a large mix of technologies and involve both people and machines. It is this broad range of complex entities that we address in this book. The basic term we use for these complex entities is *engineered systems*. However, throughout this book, when appropriate, we will freely use terms such as *products* or *services*. The term engineered systems is distinguished from *systems* in the sense that the former is created by engineers who apply science and mathematics to find suitable solutions to problems.

Traditional and high-technology manufacturing industries are responding to the challenge to satisfy consumer needs and ensure competitive and sustainable growth by reducing time to market and customizing products (or expanding product ranges) while producing the required goods in the quantities demanded with the appropriate quality at reduced costs. For instance, in the automobile sector, the lead time for manufacturing a car at the beginning of the 1990s was five to six years, whereas today it is about two to three years and is estimated to be only 18 months in the near future. Therefore, controlling schedules, costs and quality in product development, manufacturing and maintenance remains a major challenge for today's industries. Increases in complexity, decreases in development budgets and shortened time to market for new products, services and solutions are leading developers to search for new ways of improving the quality of what they deliver by improving their technologies, processes, methodologies and tools.

The overall development process is only as strong as its weakest link. A critical and largely ignored link in this process is system VVT, which comprise vital activities and involve processes. A tool of systems engineering, VVT focuses on ensuring that engineered systems are delivered as error free as possible, are functionally sound and meet or exceed the user's needs. Often VVT is carried out as merely a vehicle for finding and eliminating errors. It can do much more than that. Today, many system developers perform VVT only in the test phase of the project, a late and highly constrained period in the product development cycle. As a result, increases in overall development time and costs associated with product rework often exceed 20% of expanded engineering efforts (Capers, 1996). Admittedly, balancing testing cost and schedule with quality is difficult. However, quality problems discovered later by the user can

necessitate expensive repairs and are likely to damage the reputation of the system or, worse, damage the reputation of the system's developer.

Given the fundamental role of VVT in achieving product quality and reducing waste, this book aims at rectifying two critical current VVT problems, namely, lack of comprehensive system VVT methodology and lack of a practical, quantitative VVT process model for selecting a VVT strategy to optimize testing cost, schedule and economic risk. This book, which to a large measure is based on the European Commission–supported SysTest project, was written in order to rectify these problems.

1.1.2 Purpose

One of the central objectives of this book is the creation of generic VVT methodology. This *VVT methodology* consists of a selection of VVT activities and methods which can be applied throughout the system lifecycle in different industrial application fields and can be tailored according to the individual project needs.

The VVT methodology delivers generic means for comprehensive cost-effective VVT in the industry. In addition, the objectives of this methodology are as follows:

- To cover the entire product lifecycles from the definition to the disposal of the system
- To supply tailoring rules for different industry domains (e. g. electronics/avionics, control systems, automobile, food packaging systems, steel production), development cycles and project types
- To specify activities and methods for VVT on the system level together with their interrelationship
- To define VVT strategies that can be used in a broad variety of industrial applications

1.1.3 Intended Audience

The VVT methodology described in this book is applicable to all regional and industrial sectors. Although system VVT is performed throughout industry, it has not become a topic for research within the international community either in industry or in academia. Therefore, the definition of a generic VVT methodology will provide comprehensive knowledge for many students and practitioners. This book was written for the reader who has a background knowledge of project management, systems engineering and quality assurance. Those who participate in system development will benefit from the material covered in this book. These include:

1. *Project Managers and VVT Managers.* This book can guide project and VVT managers in the methods they select, adapt and tailor for planning, control and tracking of projects.

2. *Quality Assurance (QA)/Quality Control (QC) Staff.* For QA and, QC staff, this book offers an overview of the system QA activities and methods available and their principal advantages and disadvantages. Quality assurance staff can apply the VVT methodology guidelines for the selection of VVT procedures and the estimation of process and product risks.
3. *Members of a VVT Team.* This book serves as an aid for test teams by providing them with an overview of useful procedures for conducting a VVT process within the context of system development projects and beyond. Thus, the VVT methodology guidelines of this book become a useful tool for categorizing VVT activities within the system lifecycle overall context and by referencing further information.
4. *System Developers and Maintainers.* This book is relevant for system developers in that they deliver insight into the measures of error avoidance and error detection. Developers can draw important conclusions about the functional domains of the system developed that are critical where VVT are concerned.
5. *Mechanical, Electronics and Software Designers.* Other specialists need this book in order to take VVT aspects into account when they determine structures and select the technologies for system development, production and maintenance. This book can be an important basis for this, as it shows not only the possibilities but also the limitations of VVT procedures.
6. *Component and Subsystem Suppliers.* A clear definition and a specification with respect to VVT measures are essential, especially for system development projects that involve supplier companies. This book forms a convenient basis for those projects since it provides a mutual definition, nomenclature and techniques as well as a body of VVT methods.
7. *Auditors.* To evaluate the maturity of a development project, auditors and auditing agencies can also apply the VVT methodology. Adherence to standards, deployment of established procedures, as well as the maturity of the processes' implementation can be evaluated in this way.
8. *Regulatory and Standardization Agencies.* Material presented in this book may be helpful in forming and updating national or international standards and regulations of standardization committees in which certain procedures for defined system classes are classified as binding or just recommended. Of course, it is not the aim of this book to define or force standardization. However, it could provide important suggestions with regard to such an endeavor.

1.1.4 Book Structure and Contents

This book is divided into three parts and a set of appendices as described below.

Part I: Introduction Part I of this book contains basic introductory material organized in one chapter. It starts by describing the purpose, the intended audience, the structure and the content of the book, the scope of the applications and the terminology and notation used throughout this book. It continues by providing basic introduction to systems theory, relevant background on systems and software VVT as well as risk and uncertainty theory. In addition, this chapter introduces VVT concepts and discusses the modeling of systems and the VVT lifecycles. It then defines generic phases, views and aspects of the system lifecycle that are used in this book. Finally, the chapter provides a VVT methodology overview, typical VVT documents and a methodology for VVT tailoring.

Part II: VVT Activities and Methods Part II of this book describes the VVT activities typically associated with each phase of the system lifecycle. For each VVT activity, the book describes one or more methods for carrying out those activities:

- *Chapter 2, System VVT Activities: Development,* describes typical VVT activities which may be conducted during system development, that is, during the Definition, Design, Implementation, Integration and Qualification phases of the system's lifecycle.
- *Chapter 3, System VVT Activities: Postdevelopment,* describes typical VVT activities which may be conducted during system postdevelopment, that is, during Production, Use/Maintenance and Disposal phases of the system's lifecycle.
- *Chapter 4, System VVT Methods: Nontesting,* describes a set of VVT nontesting methods, complementing the VVT activities described in the VVT activities chapters. In particular this chapter describes the following nontesting system VVT methods: preparing VVT products, performing VVT activities and participating in reviews.
- *Chapter 5, System VVT Methods: Testing,* describes a set of VVT testing methods, complementing the VVT activities described in the VVT activities chapters. Specifically, this chapter describes a collection of system testing methods grouped into the following categories: white-box testing and black-box testing; the latter is further divided into basic testing, high-volume testing, special testing, environment testing and phase testing.

Part III: Modeling and Optimizing VVT Process Part III of this book describes ways to model system quality cost, time and risk as well as ways to acquire quality data and optimize the VVT strategy in accordance with different business objectives. In addition, Part III describes the methodology used to validate the quality models along with examples describing a system's quality improvements.

- *Chapter 6, Modeling Quality Cost, Time and Risk,* describes system quality modeling—in particular, VVT cost and risk modeling, VVT time and risk modeling and fuzzy VVT cost modeling.
- *Chapter 7, Obtaining Quality Data and Optimizing VVT Strategy,* presents typical quality data of engineered systems from various industries as well as practical ways and means to elicit and aggregate quality data (i.e., cost, time and risks of VVT activities). The chapter continues by describing various techniques to optimize VVT strategies in order to reduce cost, time and system risks.
- *Chapter 8, Methodology Validation and Examples,* describes a validation process which compares actual measurements of system quality cost and time with model prediction. Finally, this chapter provides several examples of the entire system quality improvement process.

Appendices This portion of this book contains a collection of appendices as follows:

- *Appendix A—SysTest Project*
- *Appendix B—VVT Master Plan (VVT-MP)*
- *Appendix C—Acronyms*
- *Index*

Figure 1.1 will help the reader to navigate this book.

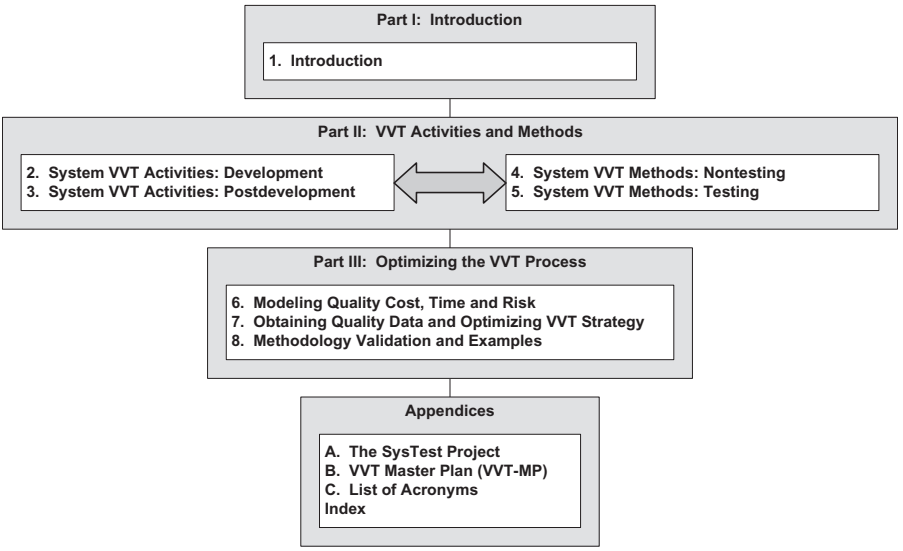

Figure 1.1 Book structure and navigation.

1.1.5 Scope of Application

This book covers system VVT, hopefully, without bias toward a specific application. The VVT methods described are applicable to a broad spectrum

of system requirements: whether safety critical or non–safety critical, whether mission critical or non–mission critical or whether the requirements are hard real time or nontemporal. The VVT methodology described herein supports the quality assurance phases all the way from system requirements definition to system disposal. Furthermore, it supports different system hierarchy levels of quality measures, from component testing to system testing. The book's VVT methodology guidelines can be applied to mass-produced systems as well as to small production quantities or few-of-a-kind paradigms.

The present book is applicable to system developments in various industrial sectors. They may be regarded as recommendations only. Or, they can be considered binding for an individual project if the stakeholders for that project agree upon this course of action.

1.1.6 Terminology and Notation

In this book, when we use the terms *has to/must*, *shall* and *should* we mean the following:

- *Has To/Must.* This is the highest level of recommendation and describes cases where the described process, procedure or approach *works only in this way*.
- *Shall.* At this level, the user is *strongly recommended* to use the described process, procedure or approach in this way.
- *Should.* This level of recommendation describes cases where this author has experienced that this process, procedure or approach is the *best*.

Each VVT activity or method described in this book is presented, as much as possible, in a common format, thus facilitating the orientation and presentation of more detailed information on each activity.

1.2 VVT SYSTEMS AND PROCESS

1.2.1 Introduction—VVT Systems and Process

This section serves as an introduction to the VVT process. It starts with the definition of an engineered system, that is, a man-made artifact that depends upon scientifically based and experiential processes that are logically applied. VVT attempts to help these systems achieve their full potential in terms of performance, efficiency and economy of precious resources. What follows is a detailed discussion of what is meant by VVT in all its manifestations. This includes a variety of definitions, as given by various experts, industries, engineering organizations and government agencies.

As a discipline VVT is an outgrowth and expansion of the earlier disciplines quality assurance and quality control. It is an evolving concept and thus will continue to be redefined with time and with the development of new techniques for design and evaluation of engineered systems. Thus, it is not surprising that there would be disagreement in the engineering and business community on just what comprises a VVT program.

Here, we attempt to give an overview of the many perceptions about VVT from the various stakeholders in the VVT process, that is, customers, manufacturers, regulators, professional organizations and government. Thus, we break down the differences between VVT definitions as seen by various technical disciplines: electrical and electronics engineering, telecommunications, artificial intelligence and the modeling and simulation community. The definitions and perceptions of VVT, as seen by the systems engineering community and more specifically by the International Council on Systems Engineering (INCOSE), are also covered, as are the VVT definitions used by the author in this book.

We attempt to give an appreciation of the difficulties of applying VVT to large and complex systems. Since VVT efforts should begin early in the lifecycles of a system and are not completed until the system is decommissioned and its components recycled, the issues are complex and manifold. Thus, we bring a section describing the stages of the system lifecycle and relate it to complementary VVT lifecycle phases.

Measuring VVT performance is key to good VVT planning. There is a delicate balance between the risks avoided by good system VVT and the risks to a system's development and deployment by too much VVT.

1.2.2 Engineered Systems

General Systems The term system (from Latin *systema*) has emerged in the twentieth century as a key building block of systems theory, an area of study that predominantly refers to the *science of systems* that resulted from Bertalanffy's general system theory (Bertalanffy, 1976).

An intuitive description of a "system" is that it is composed of separate elements organized in some fashion with certain interfaces among the elements and between the system and its environment. In addition, a system tends to affect its environment and be affected by it. This involves some type of input and output (e.g., materials, energy, information). Most importantly, a system produces results not obtainable from the collection of its individual elements.

Based on this notion, we can adopt either an elementary definition, "*A system is an interdependent group of items forming a unified whole*" (*Webster's* dictionary), or a more sophisticated definition, "*A system is a combination of components that act together to perform a function not possible with any of the*

individual parts" [Institute of Electrical and Electronics Engineers (IEEE) Electronic Terms].

Engineered Systems The goal of engineering processes is to develop and produce efficient and reliable systems (products, services or solutions) that meet a specific need under a defined set of constraints. To achieve this, the system will follow a typical creation lifecycle, whose phases could be defined as Definition, Design, Implementation, Integration, Qualification and Production. During its useful lifetime, a system will go through a Use/Maintenance phase, culminating in the disposal of the system.

According to Braha et al. (2006), the classical engineering process has several notable characteristics: (1) a search for a single solution, namely, engineers tend to seek a single solution, which often revolves around a unique design concept, for the specified problem, (2) the desire for a well-behaved system, that is, engineers prefer systems whose behavior can be predicted and encapsulated by precise description and (3) the application of a top-down problem-solving approach, which fundamentally depends on the assumption that any system can be described wholly by describing the behavior of its parts and their interactions. Therefore, according to Braha et al. (2006), classically engineered systems have the following attributes: (1) predictability, that is, the system works in predictable ways; (2) reliability, that is, the system is able to perform a required function under stated conditions for a stated period of time; (3) transparency, that is, the structure of the system and its processes can be described explicitly; and (4) controllability, that is, the system can be directly governed according to stated instructions under stated conditions.

We can now accept either the definition of the Council on Systems Engineering (INCOSE) organization: "A system is an integrated set of elements to accomplish a defined objective" adopted in 1995, or a rather sophisticated definition, attributed to Dr. Eberhardt Rechtin (1990):

> A system is a construct or collection of different elements that together produce results not obtainable by the elements alone. The elements, or parts, can include people, hardware, software, facilities, policies, and documents; that is, all things required to produce systems-level results. The results include system level qualities, properties, characteristics, functions, behavior and performance. The value added by the system as a whole, beyond that contributed independently by the parts, is primarily created by the relationship among the parts; that is, how they are interconnected.

We further accept the distinction that an engineered system is often composed of "enabling products" required to provide lifecycle support in addition to the "end products", which performs the required operational functions (see Figure 1.2). The end product may be a single manifestation of the system or may be produced in small or large quantity.

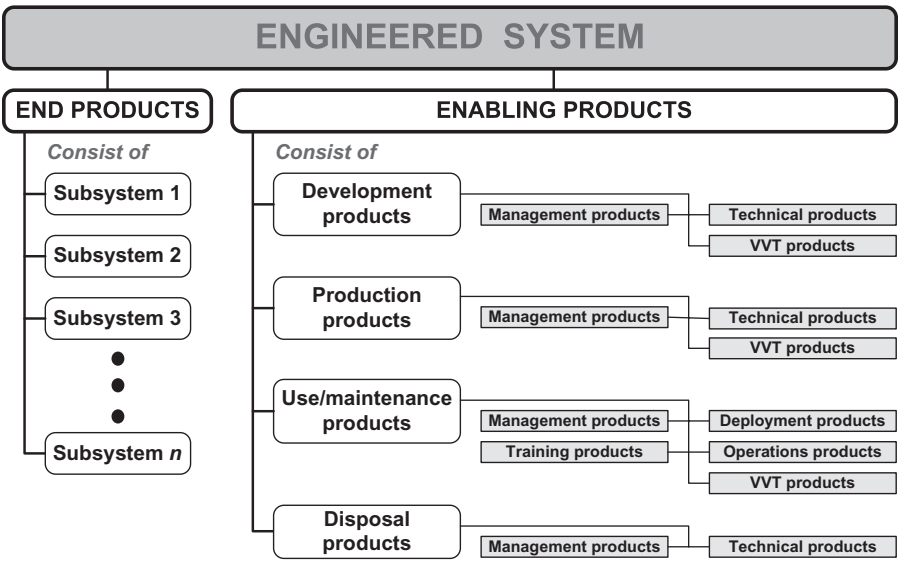

Figure 1.2 Typical structure of engineered system.

1.2.3 VVT Concepts and Definition

The acronym VVT stands for Verification, Validation and Testing. These terms have some common significance. The purpose of this discussion is to explain and encapsulate the unique meaning of each term. This section contains the following topics:

- The on-going VVT terminology debate and the general purpose of the VVT process
- The various definitions of the terms verification, validation and testing as reflected in the scientific and engineering literature
- The VVT principle and definition trends and the specific VVT definition adopted for this book

VVT Terminology and Objectives This section discusses the on-going VVT terminology debate and the general purpose of the VVT process as reflected in the scientific and engineering literature.

VVT Terminology Debate It seems that no published article on the evaluation of systems is written without first defining VVT. Many authors choose to define this term by citing some of the more popular definitions. Others, realizing the lack of clarity in those definitions, come up with their own definitions. As a result, there is confusion about exactly what VVT is and how it can be implemented in different systems.

The mere existence of confusion and the debate over definitions indicates that the VVT discipline is still in its infancy and the intent of this discussion is to dispel some of this confusion.

Purpose of the VVT Process Another question that confronts us is what should be the final purpose of the VVT process? Should it serve to eliminate errors or serve as a means to certify that a system is free of errors? Following are the arguments.

Elimination of errors is akin to debugging a computer program. The program is exercised to discover an incorrect behavior, and then the bug causing the incorrect behavior could be identified and removed. This is necessary, not only for computer programs, but also in many other fields where systems are expected to be dependable. This book reflects the author's opinion that VVT must first strive to eliminate errors if it is to be useful. On the other hand, there is a significant commercial value in being able to say that a system is free of errors and works as intended. Unfortunately, this is merely wishful thinking. To guarantee that a system is free of errors is logically impossible unless a truly exhaustive way of evaluating its functionality can be implemented. This would not be feasible for all but the most trivial systems. We conclude that the purpose of VVT should be to eliminate as many defects as possible within existing constraints of available time, money and other resources.

What is to be achieved by VVT? Fairley (1985) indicates that the goal is to assess and improve the quality of the system. He also provides quality attributes to evaluate the VVT process. These attributes, which have been altered to suit the systems arena, are presented in Table 1.1.

TABLE 1.1 VVT Quality Attributes

Function	Responding to the Following Queries
Correctness	Given valid inputs, does the system perform its tasks as expected?
Completeness	Does the system meet all of the requirements that have been placed on it?
Consistency	Are similar things handled in a similar manner? Is the system consistent with another system that is part of the same family?
Reliability	Does the system perform reasonably well in all cases, even, for instance, in the presence of pathological conditions?
Usefulness	Does the system provide a useful service?
Usability	Is the system convenient to use when carrying out its designated task?
Efficiency	Is the system efficient in its use of resources, such as time, memory, network bandwidth, and peripherals?
Standards conformance	Does the system conform to standards, both notational and external standards of interface to the outside world?
Overall cost-effectiveness	Is the system a cost-effective solution to the problem?

VVT Definitions in Various Fields The following discussion presents different definitions for the terms verification, validation and testing as reflected in the scientific and engineering literature.

1. *Nontechnical Community.* The nontechnical *Merriam-Webster's* dictionary defines the term *verify* as (1) "to confirm or substantiate in law by oath" and (2) "to establish the truth, accuracy, or reality of." It defines the term *validate* as (1) "to make legally valid," (2) "to grant official sanction to by marking," (3) "to confirm the validity of (an election)" and (4) "to support or corroborate on a sound or authoritative basis." It provides 55 different definitions for the term *test*. The most relevant nontechnical ones are (1) "a critical examination, observation, or evaluation," (2) "the procedure of submitting a statement to such conditions or operations as will lead to its proof or disproof or to its acceptance or rejection" and (3) "a basis for evaluation." The intuitive understanding of the above terms corresponds well with the nontechnical dictionary definition. The technical definition of VVT is another matter.
2. *IEEE Community.* The IEEE defines validation and verification for engineered hardware and software systems as follows (*IEEE-610*):
   - *Verification* is the process of evaluating a system or component, to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase.
   - *Validation* is the process of evaluating a system or component during or at the end of the development process, to determine whether it satisfies specified requirements.
3. *Telecommunication Community.* In its Telecom Glossary 2000, the American National Standard for Telecommunications defines the terms as follows:
   - *Verification.* (1) Comparing an activity, a process, or a product with the corresponding requirements or specifications. (2) [The] process of comparing two levels of an information system specification for proper correspondence (e.g., security policy model with top-level specification, top-level specification with source code or source code with object code).
   - *Validation.* (1) Tests to determine whether an implemented system fulfills its requirements. (2) The checking of data for correctness or for compliance with applicable standards, rules, and conventions.
   - *Testing.* Physical measurements taken (1) to verify conclusions obtained from mathematical modeling and analysis or (2) for the purpose of developing mathematical models.
4. *Artificial Intelligence Community.* Gonzalez and Barr (2000) suggest the following definitions for these terms in the Artificial Intelligence (AI) community:

- Verification is the process of ensuring that the intelligence system (1) conforms to specifications and (2) its knowledge base is consistent and complete within itself. The intent of this definition is that the process of verification represents an internal benchmark, rather than an external one. Making it internal is highly significant, as errors can be found without the need to exercise the system with test cases.
- Validation is the process of ensuring that the output of the intelligence system is equivalent to that of human experts when given the same input.

5. *Modeling and Simulation Community.* The Department of Defense (DoD) Defense Modeling and Simulation Office (DoDD-5000.59) gives a formal definition. It defines Verification and Validation (V&V) as follows:
   - Verification is the process of determining that a model implementation accurately represents the developer's conceptual description and specification.
   - Validation is the process of determining the degree to which a model is an accurate representation of the real world from the perspective of intended uses of the model.

   Balci (1998), a noted researcher in the Modeling and Simulation (M&S) field, and later Balci et al., (2000) extend the DoD definition for VVT as follows:
   - *Model verification* is substantiating that the model is transformed from one form into another, as intended, with sufficient accuracy. Model verification deals with building the model correctly. The accuracy of transforming a problem formulation into a model specification or the accuracy of converting a model representation from a micro flowchart form into an executable computer program is evaluated in model verification.
   - *Model validation* substantiates that the model, within its domain of applicability, behaves with satisfactory accuracy, consistent with the M&S objectives. Model validation deals with building an *accurate* model. An activity of accuracy assessment can be labeled as verification or validation based on an answer to the following question: In assessing the accuracy, "Does the model's behavior compare well to the corresponding system behavior?" Even if the answer to the question of accuracy is "yes," that does not answer the question of whether the model is the right one.
   - *Model testing* is determining whether inaccuracies or errors exist in the model. In model testing, the model is subjected to test data or test cases to determine if it functions properly. Test failure implies the failure of the model, not the test. A test is devised, and testing is conducted to perform either validation or verification or both. Some tests

are designed to evaluate the behavioral accuracy or validity of the model, and some other tests are intended to determine the accuracy of model transformation from one domain into another (verification). Sometimes, the whole process is called model VV&T or, for short, VVT.

VVT Concepts in System Engineering Lake (1999) explains the formal definition and intuitive meaning of V&V in system engineering (see Figure 1.3):

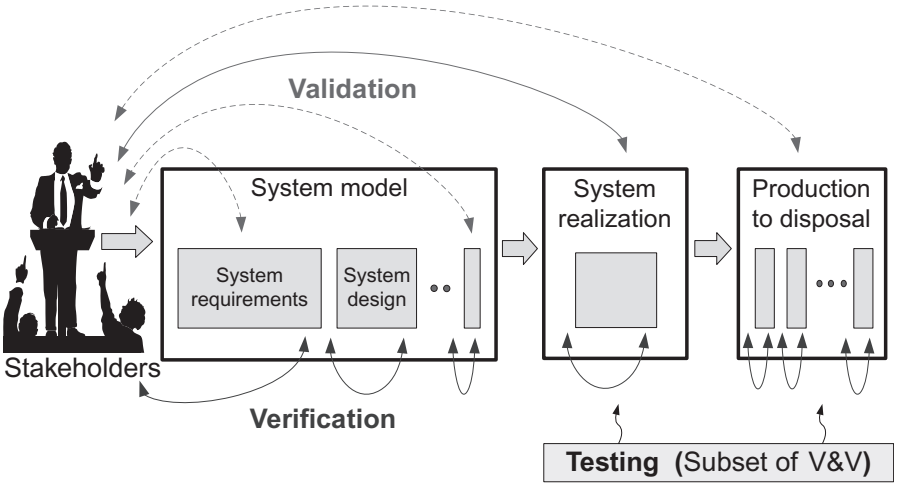

Figure 1.3 Verification and validation in system engineering perception.

- Verification is the process of evaluating a system to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase.
- Validation is the process of evaluating a system to determine whether it satisfies the stakeholders of that system.

These terms will now be further elaborated:

1. *System Verification.* The meaning of the term verification is to evaluate a realized product against specified requirements. The intent is to determine whether the finished product satisfies the specific requirements for which it was built. In addition, the verification responds to the question: "Was the product built (written, built, coded, assembled and integrated) correctly"? There are two formal definitions of verification:
   - Confirmation by examination and provision of objective evidence that the specified requirements to which a product was built, coded or

assembled has been fulfilled (American National Standards Institute/ Electronics Industries Association ANSI/EIA-632)
- The process of evaluating a system or component to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase (IEEE-610)

According to Lake (1999), verification failure (i.e., lack of confirmation) typically reveals the following types of design or implementation errors:

- Specified requirements (specifications, drawings, parts lists) have not been documented adequately.
- Developers/builders have not followed the specified requirements for the product.
- Procedures, workers, tools and equipment are improper or have been improperly used for building the product.
- Procedures and means have been improperly planned for verification.
- Verification procedures have been improperly implemented.

2. *System Validation.* The meaning of validation is evaluating a realized product against specified (or unspecified) requirements in order to determine whether the product satisfies its stakeholders. In other words, validating a product is determining whether the product does what it is supposed to do in the intended operational environments. In addition, the validation responds to the question: "Was the right product built?" There are two formal definitions of the term validation:
   - Confirmation by examination and provision of objective evidence that the specific intended use of a product (developed or purchased), or aggregation of products, is accomplished in an intended usage environment (ANSI/EIA-632)
   - "The process of evaluating a system or component during or at the end of the development process to determine whether it satisfies specified requirements" (IEEE-610)

According to Lake (1999) typical validation errors stem from:

- Input requirements not adequately identified
- Design process incorrectly executed
- Input requirement changes not communicated
- Procedures and means improperly planned for validation
- Validation procedures improperly implemented

3. *System Testing.* The meaning of the term *testing* is operating or activating a realized product or system under specified conditions and observing or recording the exhibited behavior. Here are two formal definitions of this term:

- "An activity in which a system or component is executed under specified conditions, the results are observed or recorded, and an evaluation is made of some aspect of the system or component" (IEEE-610)
- "The process of operating a system or component under specified conditions, observing or recording the results, and making an evaluation of some aspect of the system or component" (IEEE-610).

VVT Definition in This Book This section concludes this VVT presentation. It provides the author's view as to the trends in VVT definitions. These trends form the basis for the VVT definition which has been adopted for this book.

1. *Trends in VVT Definitions.* It should by now be obvious that we really do not have a single concept regarding the meaning of the VVT of systems, at least from the standpoint of the technical community. Some say that validation and verification are one and the same thing, others say verification deals with specifications, others say it is validation that deals with specifications while still others say that they both do. Furthermore, some authors relate consistency and completeness to verification while others do so with validation. Nevertheless, some trends have emerged (see Table 1.2). These trends are not universally accepted but simply were observed.

TABLE 1.2 Trends in VVT Definition

Trend Number	Description
1	Verification deals with satisfying the written specifications of systems.
2	Verification involves the internal structural correctness of systems.
3	Verification relates to the evolving lifecycle processes of systems.
4	Validation compares the system to the needs of stakeholders. These needs may vary in time.
5	In order to validate a system, the requirements of the stakeholders, whether formally specified or not, must be known.
6	Testing involves some type of exercising the system. This is a static and dynamic process that evaluates functional correctness.
7	Testing can be accomplished as a subset of either verification or validation.

2. *Principles of VVT.* Balci (1998) suggests a set of principles for carrying out verification and validation properly. This information, in a condensed form, is provided in Table 1.3 with some adjustments to account for the systems environment.

TABLE 1.3 Principles of VVT

Principle Number	Description
1	VVT has to be conducted throughout the entire system lifetime and faults should be detected as early as possible in the system life.
2	VVT has to be planned, documented and conducted by unbiased parties.
3	Performing complete system VVT is not possible and a successful VVT of each subsystem does not imply overall system credibility.

3. *VVT Definition in This Book.* This book has adopted the systems engineering VVT definition based on the 15 VVT principles suggested by Balci (1998). Specifically, this is the collection of VVT definitions set forth in IEEE-610 and elaborated upon by Lake (1999) (see Table 1.4). The general acceptance of these definitions by the system engineering community was a factor in this decision.

TABLE 1.4 VVT Definition in This Book

Term	Definition
Verification	**The process of evaluating a system to determine whether the products of a given lifecycle phase satisfy the conditions imposed at the start of that phase.**
Validation	**The process of evaluating a system to determine whether it satisfies the stakeholders of that system.**
Testing	**An activity in which a system is activated under specified conditions, the results are observed or recorded, and an evaluation is made of some aspect of the system.**

1.2.4 The Fundamental VVT Dilemma

It is well understood that it is impossible to prove that a system actually meets all it functional capabilities as well as all standards, statuary directives, and ethical values and at the same time adheres to business objectives. The main limiting factors other than plain physics are the cost and time to market, which is required in order to bring products into common use. Therefore it is the domain of the system VVT engineer and management to strive for an optimal solution of the VVT process. As this issue is a central theme in system VVT, the book addresses the issues of cost, risk and time of the VVT process in great detail. Figure 1.4 depicts the fundamental balancing and optimizing of the VVT process. Highlighted are the business objectives emphasized in this book.

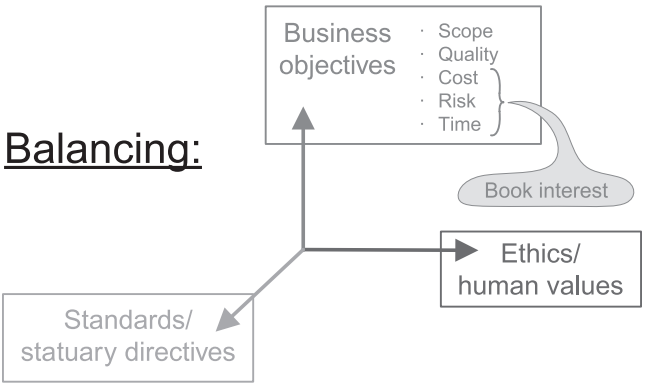

Figure 1.4 Balancing and optimizing the VVT process.

1.2.5 Modeling Systems and VVT Lifecycle

This section describes major system lifecycle models and in particular systems' lifecycle definitions used by U.S. government and commercial organizations. A generic system lifecycle adopted for this book is also presented.

Major System Lifecycle Models An overall system lifecycle model describes a cradle-to-grave paradigm of engineered systems. Different organizations [e.g., the National Aeronautics and Space Administration (NASA), DoD] and industries (e.g., automobile, electronics, telecommunication, aerospace) define various system lifecycle models. For example, the DoD acquisition lifecycle process has 4 major phases and 22 minor phases, as defined in Table 1.5.

TABLE 1.5 Major System Lifecycle Phases as Defined by U.S. DoD

Major Systems Lifecycle Phase			
0	**I**	**II**	**III**
Concept Exploration **(CE)**	Program Definition & Risk Reduction **(PD&RR)**	Engineering & Manufacturing Developmen **(EMD)**	Production, Fielding/Deployment & Operational Support **(PFD&OS)**
1. System analysis	6. Concept design update	11. Detail design	17. Production rate verification
2. Requirements definition	7. Subsystem trade-off	12. Development	18. Operational test & evaluation
3. Conceptual design	8. Preliminary design	13. Risk management	19. Deployment
4. Technology & risk assessment	9. Prototyping, test, & evaluation	14. Development test and evaluation	20. Operational support & upgrade

TABLE 1.5 ***Continued***

Major Systems Lifecycle Phase			
0	**I**	**II**	**III**
5. Preliminary cost, schedule & concept	10. Integration of manufacturing & supportability considerations	15. System Integration, test & evaluation	21. Retirement
		16. Manufacturing process & verification	22. Replacement planning

0. *Concept Exploration.* The CE phase begins with a definition of project or product objectives, mission definition, definition of functional requirements, definition of candidate architectures, allocation of requirements to one or more selected architectures and concepts, trade-offs and conceptual design synthesis and selection of a preferred design concept. An important part of this phase is the assessment of concept performance and technology demands and the initiation of a preliminary risk management process.

I. *Program Definition and Risk Reduction.* The PD&RR phase is oriented to a risk management strategy in order to prove that the system will work prior to committing large amounts of resources to its full-scale engineering and manufacturing development. This is the first phase in the development cycle where significant effort is allocated to developing tangible products such as top-level specifications, decomposing and allocating system requirements and design constraints to lower levels, supporting preliminary design, monitoring integration of subsystem trade-offs and designs and detailed project plans.

II. *Engineering and Manufacturing Development.* During the EMD phase, detailed design and test of all components and the integrated system are accomplished. This may involve fabrication and testing of engineering models and prototypes in order to check that the design is correct. The hardware and software design for the EMD usually differ from those of the PD&RR phase. This is usually justified to minimize the PD&RR phase costs and to take advantage of lessons learned during PD&RR in order to improve the EMD design. Thus, most of the analysis, modeling, simulation, trade-off and synthesis tasks performed during CE and PD&RR are repeated at a higher fidelity. A requirement validation process should be conducted before the EMD hardware and software is produced. This will ensure that the entire system will function as envisioned.

III. *Production, Fielding/Deployment and Operations and Support.* During production, deployment and operational use, the focus is on solving

problems that arise during manufacturing, assembly, integration and verification as well as the transition into its deployed configuration. Additionally, attention is given to customer orientation, validation and acceptance testing. During the phase of operations and support, systems are usually under the control of the purchasers/operators. This involves a turnover of the system from experienced developers into less experienced operators. This leads to a strong operations and support presence by the developers in order to train and initially help operate the system. During this period, there may be upgrades to the system to achieve higher performance levels.

Government and Commercial Program Phases INCOSE (2007) further illustrates and compares several typical lifecycle phases of government and commercial organizations (see Figure 1.5). This figure emphasizes that system lifecycles in different domains are fundamentally similar in that they move from requirements, definition, and design through manufacturing, deployment, operations and support (and sometimes to deactivation), but they differ in the vocabulary used and nuances within the sequential process.

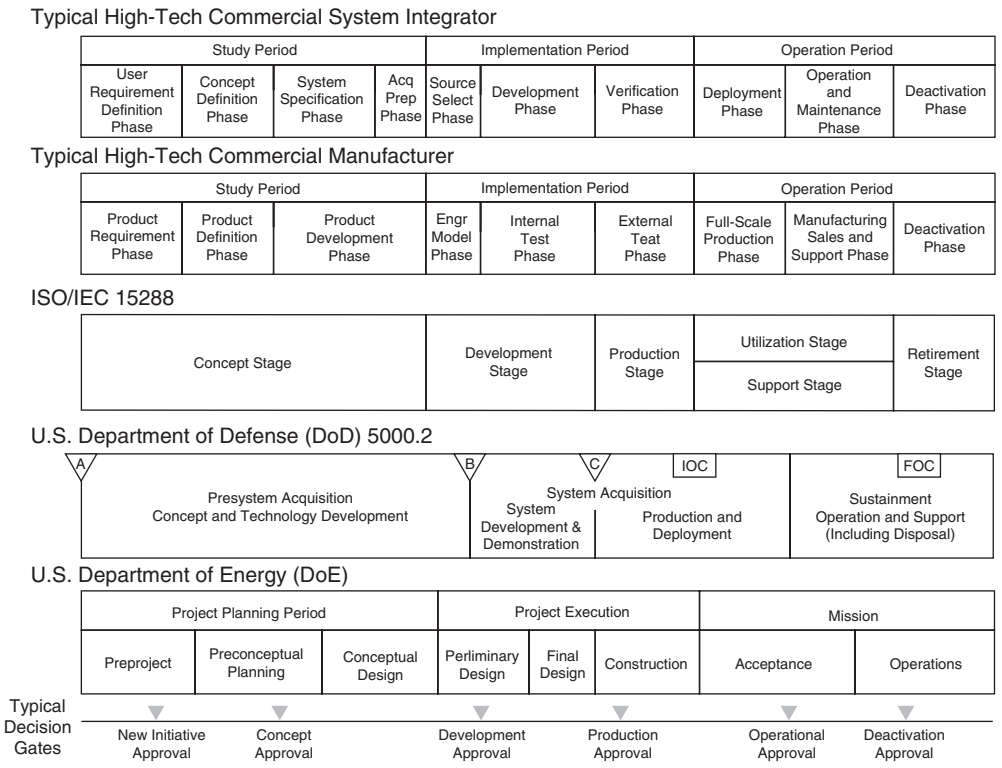

Figure 1.5 System lifecycle phases as illustrated in INCOSE, 2007.

Generic System Lifecycle Adopted for This Book This book has adopted the generic system lifecycle model (see Table 1.6) that is used in the SysTest project due to its generality and practicality. It is a generic extension of the model of system lifecycle phases and VVT activities suggested by Addy (1999) and Boehm (2001). This system lifecycle model extends the well-established V-Model (Martin and Bahill, 1996), which portrays project evolution during the development portion of the system lifecycle.

TABLE 1.6 Generic System Lifecycle Definition Model

Phase	Purpose
	Development
Definition	Formulate the system operational concepts and develop the system requirements.
Design	Create a technical concept and architecture for the system.
Implementation	Create the elements of the system. Each element is built or purchased, then tested to ensure its stand-alone compliance with its allocated requirements.
Integration	Connect the implemented elements into a complete system.
Qualification	Perform formal and operational tests on the completed system to assure the quality of the system as a whole.
	Postdevelopment
Production	Produce the completed system in appropriate quantities.
Use/Maintenance	Operate the system in its intended environment in order to accomplish intended functionality, maintain the system and correct any defects.
Disposal	Properly dispose of the system and its elements upon completion of its life.

Figure 1.6 depicts the V-Model as a part of the overall generic system lifecycle model developed during the SysTest project and adopted for this book (Engel et al., 2001).

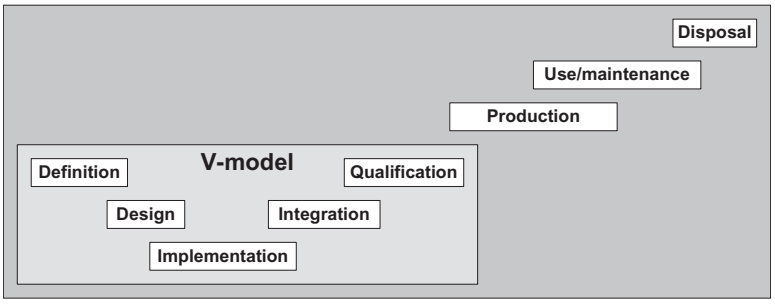

Figure 1.6 V-Model as part of overall generic system lifecycle model.

The left-hand side of the V-Model corresponds to satisfying stakeholders' requirements and the design of the desired system and its components. The right-hand side of the V-Model consists of building the individual components, integrating them and then verifying and validating the whole system. Figure 1.6 depicts the V-Model as a part of the overall generic system lifecycle model developed during the SysTest project and adopted for this book (Engel et al., 2001). Figure 1.7 depicts a generic system lifecycle model together with the corresponding generic VVT lifecycle, with which it is associated.

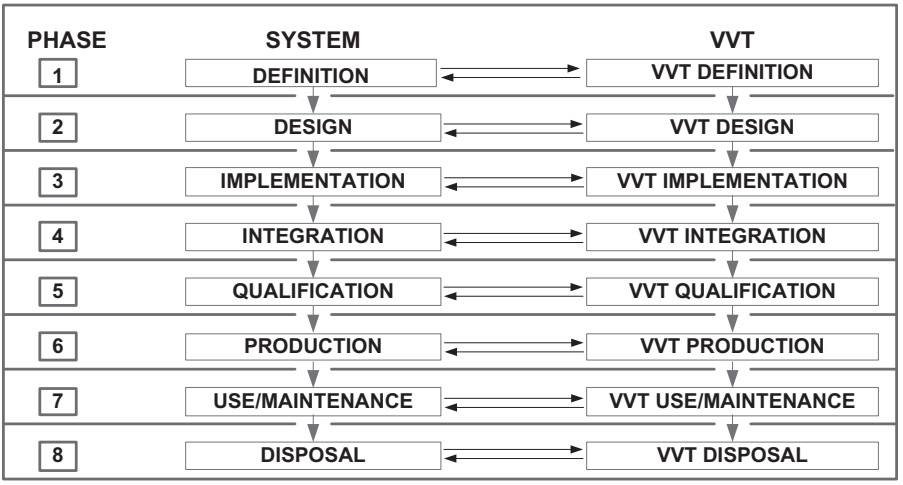

Figure 1.7 Modeling generic systems and VVT lifecycles.

1.2.6 Modeling VVT and Risks as Cost and Time Drivers

Traditional Modeling Quality Cost The cost of quality is the overall cost associated with ensuring the quality of products or services delivered to customers. In the 1950s, Joseph M. Juran developed his cost-of-quality concepts (see Juran and Gryna, 1980). Later, several researchers (e.g., Montgomery, 2001) encapsulated a lexical qualitative model of cost of quality. Some researchers augmented the information with field-obtained quality cost data (e.g., Sörqvist, 1998). Due to the relevancy and fundamental nature of this qualitative cost-of-quality model, it is presented below with relevant alterations emanating from the perspective of this book. Specifically, the cost of quality in manufacturing and service industries is composed of four components: (1) prevention cost such as quality planning and training, (2) assessment cost such as product inspection and testing, (3) internal failures

cost such as scrap, rework and retest and (4) external failure costs such as warranty charges, liability cost and indirect cost. We will now map system quality costs to this model.

1. *Prevention Costs.* Prevention costs are costs expanded on the prevention of nonconformance to specifications during system development, manufacturing and maintenance. Important subcategories of prevention costs are shown in Table 1.7.

TABLE 1.7 Subcategories of Prevention Cost

Subcategories	Type
Quality Planning. Costs associated with the creation of various quality plans (e.g., inspection plan, reliability plan).	VVT cost
Product/Process Design. Costs incurred during the quality evaluation of system development and production processes which are intended to improve the overall quality of products as well as costs incurred during the evaluation of the development and manufacturing effectiveness (e.g., input versus output, return on investment)	VVT cost
Process control. The cost of process control activities, such as collecting samples and generating control charts which monitor the development or the manufacturing process in an effort to reduce variation and create quality within system.	VVT cost
Burn-in. The cost of preshipment exercising and evaluation of system in order to minimize early-life defects in the field.	VVT cost
Training. The cost of developing, implementing, operating, and maintaining training programs in order to achieve system quality.	VVT cost
Quality Data Acquisition and Analysis. The cost associated with creating, purchasing, and operating quality of data collection and distribution system as well as the cost of running the quality data system to obtain information about systems and process quality performance and analyzing and publishing it for management, customers and other stakeholders.	VVT cost

2. *Assessment Costs.* Assessment costs are those costs associated with measuring and evaluating purchased materials, components and subsystems as well as verifying, validating and testing systems (i.e., end products and enabling products) to ensure conformance to specified requirements and standards. The major subcategories of assessment costs are described in Table 1.8.

TABLE 1.8 Subcategories of Assessment Cost

Subcategories	Type
Inspection and Test of Incoming Material. Costs associated with the inspection and testing of appropriate vendor's supplied raw material, components and subcategory either at the vendor's facility or at the receiving station of the firm. In addition, this subcategory includes verification of all vendor-supplied documentation as well as periodic audit of the vendor's quality assurance system.	VVT cost
Systems Verification, Validation and Test. The cost of checking the conformance of the systems throughout the various stages of development and manufacturing, including final acceptance testing, packing and shipping checks and any test done at the customer's facilities prior to turning systems over to the customer. In general, assessment cost also covers tests and evaluation associated with system maintenance activities as well as verification and validation of appropriate disposal process.	VVT cost
Consumed Materials and Products. The cost of material and products consumed in destructive quality tests or devalued by reliability tests.	VVT cost
Maintaining Accuracy of Test Equipment. The cost of ensuring that the measuring instruments and equipment are calibrated on an ongoing basis.	VVT cost

3. *Internal Failure Costs.* Internal failure costs are incurred when materials, components, subsystems or systems do not meet quality requirements and these failure are discovered prior to delivery of the systems to customers. The major subcategories of internal failure costs are described in Table 1.9.

TABLE 1.9 Subcategories of Internal Failure Cost

Subcategories	Type
Scrap. The net loss of labor, material and overhead resulting from defective product or systems that cannot economically be repaired or used.	Risk cost
Rework. The cost of correcting system chronic or sporadic defects so that they meet specifications. This process may transpire once or several times.	Risk cost
Retest. The cost of repeated verification, validation and testing of systems that have undergone rework or other modifications.	Risk cost
Failure Analysis. The cost incurred to determine the global causes of recurring system failures. Note that this subcategory is not referring to a regular testing process but to a wider phenomenon of persistent system failures.	Risk cost

TABLE 1.9 ***Continued***

Subcategories	Type
Downtime. The cost associated with idle development or production facilities and manpower that result from nonconformance to requirements. The development may be halted until certain information is obtained. A production line may be down while a defective system or product is evaluated or repaired.	Risk cost
Yield Losses. The cost of process yield that is lower than might be attainable by improved quality controls.	Risk cost
Downgrading. The cost associated with inferior products and systems that do not meet the entire customer's requirements. Downgrading implies that such products yield less profit relative to products that conform to specifications. In addition, inferior products adversely affect the reputation of the firm, causing loss of revenues.	Risk cost

4. *External Failure Costs.* External failure costs occur when systems do not perform satisfactorily and the problems are identified after these systems have been supplied to customers. The subcategories of external failure costs are described in Table 1.10.

TABLE 1.10 Subcategories of External Failure Cost

Subcategories	Type
Complaint Adjustment. All costs associated with the investigation and adjustment of either justified or not justified complaints attributable to the nonconforming product.	Risk cost
Handling Defective Products and Systems. All costs associated with either fixing systems at customers' premises or replacing nonconforming products and systems that are returned from the field.	Risk cost
Warranty Charges. All costs involved in service to customers of faulty systems under warranty contracts.	Risk cost
Liability Costs. All costs associated with defective products and systems incurred as a result of system liability litigations.	Risk cost
Indirect Costs. Costs incurred because of customer dissatisfaction with the level of quality of the delivered system. They include the costs of business reputation loss, future business loss and market share loss that may result from delivering defective systems that do not meet the customer's expectations.	Risk cost

Waste in Product Development The Lean Aerospace Initiative (LAI) was born out of declining defense budgets and military industrial overcapacity, prompting a new defense acquisition paradigm, that is, affordability rather than performance. The U.S. Air Force (USAF) and the Massachusetts Institute of Technology (MIT) launched this initiative in 1993.

Researchers dedicated to the philosophy called "lean" are interested in eliminating waste that occurs during systems' development phase of projects.

Womack and Jones (2003) classified all product-making activities into Value Adding (VA), to be continually perfected; Non–Value Adding (NVA), to be eliminated; and Required Non–Value Adding (RNVA), such as those required by contract or law, to be faithfully executed. No formal study is available on the relative amounts of NVA and RNVA waste in the aerospace programs (Oppenheim, 2004). Table 1.11 shows two sets of product development waste categories as classified by two studies.

TABLE 1.11 Two Sets of Product Development Waste Classifications

Classification by Millard (2001)	Classification by Morgan (2002)
1. Overproduction (creating unnecessary information)	1. Hand off (transfer of process between parties)
2. Inventory (keeping more information than needed)	2. External quality enforcement (including performance requirements)
3. Transportation (inefficient transmittal of information)	3. Waiting
4. Unnecessary movement (people having to move to gain or access information)	4. Transaction waste
5. Waiting (for information, data, inputs, approvals, releases, etc.)	5. Reinvention waste
6. Defects (insufficient quality of information, requiring rework)	6. Lack of system discipline
7. Overprocessing (working more than necessary to produce the outcome)	7. High process an arrival variation
	8. System overutilization and expediting
	9. Ineffective communication
	10. Large batch sizes
	11. Unsynchronized concurrent processes

In an ideal world, systems are created perfectly and VVT procedures would not be necessary. Therefore, performing VVT and incurring VVT appraisal and impact risks are clearly NVA activities. Obviously, optimizing the VVT strategy leads to less costly NVA results. Our world is not ideal and the VVT process is a necessary expenditure that is required to ensure the quality of systems. Therefore, one can say that just about all VVT activities lie on the border between VA and NVA activity regions.

Modeling Cost and Risk VVT cost can be considered a cost associated with classical prevention and assessment, while risk impact cost is usually associated with sustaining internal and external failures. Developing risk-based cost models involves three activities:

- Identifying VVT risks
- Estimating risk probability
- Estimating risk effects

In the literature, we find several methodologies dealing with these topics. The main ones are discussed below.

Methodology Based on Perception of Engineering Process A detailed approximation of the underlying cost and risk of a project can be obtained by viewing the engineering process as a tree structure and each node in the tree is an engineering activity. The standard engineering tool of Work Breakdown Structure (WBS) is an available vehicle to promote and support this methodology. Engineering process parameters such as cost/duration, including the VVT tasks, are first identified. Experts then assign valuations to them based on the experts' technical knowledge. To take into account uncertainties, rather than assigning only a best estimate of task cost and duration, these experts can assign a minimum, a most likely and a maximum estimate for each of these two quantities.

VVT activity costs and durations are fairly easy to predict, whereas the costs and durations of engineering processes are somewhat less predictable due to their physical nature. Fortunately, engineering experts are able to do a fairly good job at estimating risks, risk impact probabilities, and risk impact costs. Because expert opinions often differ, the cost estimates for normal engineering activities and the risk cost estimates are recognized to be probability functions across the different categories and expert opinions. The data are presented to participants and stakeholders as a range of values rather than a single value in terms of a cost–risk curve (e.g., a histogram of risk–cost density distribution). It should be noted that more sophisticated approaches for transforming the three estimate levels into probabilistic data are available, for example, with the aid of a beta distribution (Fente et al., 1999).

Methodology Based on Balancing Cost/Availability and Benefits Browning (1998, 1999) describes a method for identifying acceptable risks. The method balances product pricing and availability timing with the value of the product to the customer. The designers of systems must fit the design process to optimize this process. Browning's thesis first addresses the sources of risk of not meeting this optimization and classifies it into six categories: (1) cost, (2) schedule, (3) performance, (4) technology, (5) business and (6) market risks. Then he builds a framework and a model to represent the relationships between these risks. A stochastic simulation is then used to generate probability distributions of possible costs, schedules and performance outcomes. These distributions model uncertainty and are analyzed in relation to impact functions. The model provides the means to explore several management options for optimizing the above parameters.

Methodology Based on Holistic Philosophy of Risk Scenarios Haimes (1998) coined the term Hierarchical Holographic Modeling (HHM) to depict complex systems using multiple models created along different perspectives. Extending this concept, Haimes et al. (2002) proposed an analytic framework called Risk Filtering, Ranking, and Management (RFRM), which can identify, prioritize, assess, and manage risk scenarios of large-scale systems. In a nutshell, the risk assessment portion of RFRM follows these steps: First, the HHM must be developed to describe a multifaceted model of the system's "as-planned" scenario. Then, the set of risk scenarios is qualitatively filtered

and ranked according to the system stakeholders' views. Finally, a quantitative filtering and ranking of possible risks must be carried out based on the likelihood of system failures and the consequences of such events. Lamm and Haimes (2002) use the HHM and RFRM methodologies to analyze the security of the U.S. national information infrastructures.

Methodology Based on System Safety Program Requirements Muessig et al. (1997) describe another methodology in the context of a risk–benefit analysis approach to the selection of an optimal set of Verification, Validation, and Accreditation (VV&A) activities. This risk modeling is based on an adaptation of the U.S. military standard MIL-STD-882C, System Safety Program Requirements. In the model, VVT risks are quantified in terms of probability of occurrence and impact or severity levels within the context of specific applications. Two variables are involved in modeling risks as cost drivers: (1) the uncertainty of risk occurrence and (2) the severity of risk impact.

1. *Uncertainty of Risk Occurrence.* The first element affecting risk is the uncertainty with which undesirable events occur. The risk model defines the probability of occurrence of a given risk factor in different ways, depending on the category of the risk factor that is being considered. The effect of undesirable events impacting the system can be measured by (1) the number of items affected in a population, (2) the number of events per unit of time or (3) the total number of events over the life of the system or product.

   The model of Muessig et al. (1997) divides the probability continuum into five bands and gives guidelines for selecting the appropriate band. Table 1.12, extracted from MIL-STD-882C, provides these guidelines in terms of the number of undesirable events over a lifetime and per number of items in a population.

TABLE 1.12 Probability of Risk Occurrence

Probability Description	Likelihood of Occurrence over Lifetime of Item	Likelihood of Occurrence by Number of Items
Frequent	Likely to occur frequently	Widely experienced
Probable	Will occur several times in life of item	Will occur frequently
Occasional	Likely to occur sometime in life of item	Will occur in several items
Remote	Unlikely but possible to occur in life of item	Unlikely but can reasonably be expected to occur
Improbable	So unlikely it can be assumed occurrence may not be experienced	Unlikely to occur but possible

The reader may substitute "system" or "product" for the word "item," as appropriate.

2. *Severity of Risk Impact.* The second element affecting risk is the severity of the impact of an undesirable event, should the event be experienced. The risk model developed by Muessig et al. (1997) expands the MIL-STD-882C while grouping the impact severity into four bands: catastrophic, critical, marginal and negligible. The criterion for assigning one of these impact bands to a particular risk depends on the category of that risk. The impact categories that are discussed in the model are personnel and equipment safety, environmental damage and occupational illness. Depending on the particular use of the system being considered, some of these impact categories might not apply, and additional categories might be added—for example, impact on end-user capability or effectiveness, cost, performance, schedule and political or public reaction. A set of criteria for determining the level of impact for each of the different impact categories is provided in Table 1.13 as an illustrative guideline.

TABLE 1.13 Severity of Risk Effects

	Risk by Impact Levels			
Categories	Catastrophic	Critical	Marginal	Negligible
Human safety	Death	Severe injury	Minor injury	Less than minor injury
Systems safety	Major equipment loss; broad-scale major damage	Small-scale major damage	Broad-scale minor damage	Small-scale minor damage
Environmental damage	Severe	Major	Minor	Some trivial
Occupational illness	Severe and broad scale	Severe or broad scale	Minor or small scale	Minor and small scale
Financial losses of program	Loss of program funds; 100% cost growth	Fund reductions; 50–100% cost growth	20–50% cost growth	<20% cost growth
Functional performance of product	Design does not meet critical thresholds	Severe design deficiencies but thresholds met	Minor design flaws but fixable	Some trivial "out of spec" design elements
Schedule slippage of product	Slip reduces overall capabilities	Slip has major cost impacts	Slip causes internal turmoil	Republish schedules

TABLE 1.13 ***Continued***

Categories	Risk by Impact Levels			
	Catastrophic	Critical	Marginal	Negligible
Political or public impact of event	Impact widespread (Watergate)	Significant (Tailhook '91)	Embarrassment ($200 hammer)	Local
Negative impact due to unidentified stakeholders	Major stakeholder blocks program (Israeli AWACS sale to China)	Stakeholder requires product modifications (FAA disqualifies new aircraft)	Stakeholder requires minor system modifications	Upgrading sales campaign to cover newly recognized stakeholders
Future losses of potential revenues	Customers determined to abandon product	Major market share loss	Customers dissatisfied with product	Competitor plan to develop similar product

1.3 CANONICAL SYSTEMS VVT PARADIGM

1.3.1 Introduction—Canonical Systems VVT Paradigm

An engineered system does not appear suddenly in just an instant. Like any other entity, it needs to be brought into being, cared for and nourished, challenged and utilized and finally put to rest. Thus, the concept of a system life is appropriate. This section discusses that life and describes the role of VVT in its phases. This is presented in terms of the canonical system VVT paradigm composed of (1) phases of the systems lifecycle, (2) views of the systems and (3) aspects of the systems.

A system, in this context, is a set of interacting or interdependent entities, man made or otherwise, existing and forming an integrated whole that fulfills a certain purpose or set of objectives. For an engineered system to adequately meet its objectives, the goal should be to invent, develop, adapt or optimize system behavior within a set of required properties. The man-made parts of an engineered system can undergo development from different disciplines, such as mechanics, hydromechanics, electronics, computation and programming. Other parts, such as human operators or technicians, can also undergo development from other disciplines, such as education, training and work experience.

Figure 1.8 helps the reader to envisage the many interactions involved in the VVT process. It depicts the canonical system VVT paradigm as a three-dimensional object:

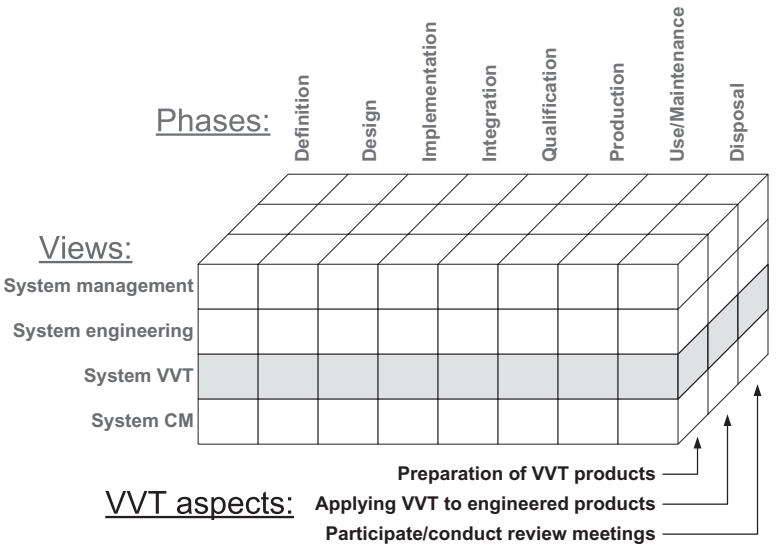

Figure 1.8 Canonical system VVT paradigm.

- *First Dimension.* Lifecycle phases include all the system lifecycle phases (i.e., Definition to Disposal).
- *Second Dimension.* System views include, among others, the following components: System management, Systems engineering, System VVT and System Configuration Management (CM).
- *Third Dimension.* Aspects of systems include the following components: Preparation of VVT products, Applying VVT to engineered products and Participating or conducting reviews.

Knowing the phases of the system lifecycle is essential for understanding how VVT is implemented throughout the life of a system. Thus, each phase is discussed separately and the appropriate VVT activities for that phase are described. During the entire lifecycle, from system definition to system disposal, there are at least four views of the system. Naturally, the most important view for this book is VVT. For completeness, short descriptions of the remaining views are also provided.

Here each activity of a system lifecycle can be categorized by placing each of them in one of the cubes depicted in the three-dimensional stack of cubes shown. These activities describe what has to be done in order to achieve the desired degree of quality in a system.

The VVT activities, however, indicate only what may be done to assure the quality of a system. Thus, for each VVT activity, this book provides one or more VVT implementation methods. These VVT methods describe how to perform an activity by defining a sequence of steps that should be performed.

From this perspective, a step within a method may indeed be a VVT activity unto itself. While some VVT activities are straightforward and may be implemented by only one method, others may be carried out using one of several methods. An example of a hierarchy depicting activities and methods is shown in Figure 1.9. Each element of the canonical system VVT paradigm (i.e., phases of the system lifecycle, views of the system and aspects of the system) will now be discussed in more details.

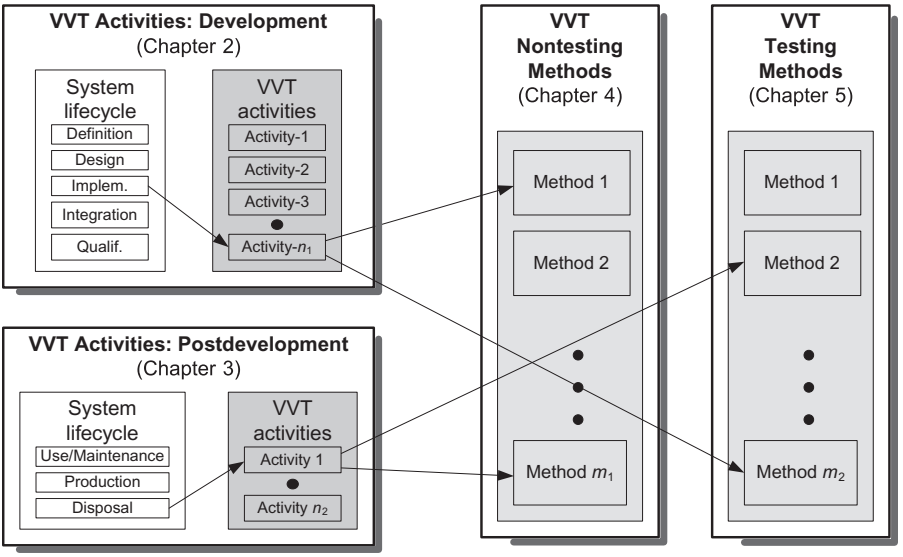

Figure 1.9 Hierarchy of VVT activities and methods.

1.3.2 Phases of the System Lifecycle

Each individual activity of a system lifecycle is allocated to one of the phases and works smoothly together with other activities to achieve the overall goals of that phase. There are several mostly overlapping phases, each describing a particular period of the overall system lifecycle. Depending on the system (hardware versus software development, safety-critical versus noncritical application, etc.), some of these phases are considered more relevant than others. As mentioned above, the canonical phases of a system's lifecycle are Definition, Design, Implementation, Integration, Qualification, Production, Use/Maintenance and Disposal.

In our system lifecycle framework, eight phases encompass the system lifecycle. Depending on the system under consideration, some of these phases may be more or less important. These eight phases pretty much cover the same areas as the five phases called out in the ISO/IEC 15288: Concept (Define/Design), Development (Implement/Integrate/Qualify), Production (Produce),

Utilization and Support (Use and Maintain) and Retirement (Disposal). The eight phases of a system lifecycle are described in the following.

System Definition During the system Definition phase, the requirements of the system are elaborated as completely and precisely as possible in terms of system, hardware and software requirements. Specifications that could constitute the actual system definition could take many forms. For instance, textual requirements, formal requirements, system models or prototypes can be artifacts of system requirements activity.

From the perspective of VVT, during this phase, a project should produce a set of system requirements that are complete, clear and consistent. VVT planning consists of defining forward-looking VVT-related concepts and goals. Specific details of VVT are few, but the planner should be looking at defining the overall VVT framework in general terms that support the emerging system architecture. For example, if the system requirements mandate built-in test capabilities, the VVT philosophy could emphasize intrinsic self-instrumentation capabilities within components in order to reduce the need for developing intrusive and expensive instrumentation.

In the Definition phase, allocation of requirements to hardware and software is usually incomplete; so many specifics of VVT cannot be fully developed. Once systems engineering begins to define the Technical Performance Measures (TPMs) that will assist in meeting system performance requirements, some of the details of VVT requirements can be established. The VVT philosophy during this phase must be forward looking and flexible, as this is the time that system definition is most fluid.

The primary objective in VVT planning in this phase is to define the framework for VVT throughout the program to the level of detail possible. Just as the system receives its architectural concepts during this phase, VVT develops its own architecture that supports the program needs. As system requirements are being analyzed and lower level specifications are being written, VVT planning focuses on the analysis of test requirements and influence of specifications from a test and instrumentation perspective. If self-test requirements are articulated at a top level, or if requirements analysis and derivation imply the need for self-instrument requirements, then the VVT planning can both influence and build upon these expected capabilities as they become defined.

System Design The technical concept of the system, the principles and the underlying system architecture for the implementation of the system are determined during the system Design phase. The total complex system is divided into manageable subsystems and components and the functions of the individual elements as well as their interrelations are described.

As requirements get refined and assigned into subsystems and components, VVT will now have a more concrete structure against which to direct specific test strategies. General TPMs will become allocated and apportioned to sub-

systems and components. The resulting greater specificity allows VVT planning efforts to be directed toward the implementation phase and integration phase needs.

System Implementation The design concept is realized during the system implementation phase. If the system is a hardware-based system, this implementation is only a prototype (i.e., the first instance of the system built) that must be reproduced during the system Production phase. At the completion of the system Implementation phase, all individual components of the overall system should be available and functioning.

During system implementation, VVT efforts are directed toward those emerging subsystems, their verification against system requirements and their refinement. As requirements are verified with respect to implemented components, they should also be validated against stakeholder needs. This validation should be a continuous process. Whenever subsystem or component definition and specificity permit, the associated requirements should be validated.

System Integration The focal point of this phase is the integration of the implemented subsystems with the aim of setting up the complete system.

VVT activities during system integration are directed at verifying that the interfaces between subsystems or components as well as between the system as a whole and external elements meet requirements and that the whole meets system requirements as well. VVT activity should also be focused toward validation of each requirement within the relevant integrated subsystem. VVT planning during this phase is directed toward preparing for qualification of the system.

System Qualification The system Qualification phase is a formal phase during which the system runs through a number of tests often prescribed by external agencies, customers or standards. The goal is to assure the quality of the system as a whole. Ideally, during this phase, no constructive developments on the system should be carried out. In practice, however, often certain parts of the system are being tested while other parts are still under various stages of development.[1]

At this point, the formal validation of the verified requirements ensures that the system meets the stakeholder true needs and that those needs are accurately reflected in the captured requirements. VVT activities include testing the system and ensuring that all requirements are verified using the proper method (i.e., analysis, inspection, demonstration, testing or certification). VVT planning consists of selecting appropriate qualification testing for inclusion in the Production phase as a subset of acceptance testing. VVT planning starts the preparations to support testing of purchased parts and conduct-

[1]Concurrent engineering is a methodology of developing different parts of a system in an unsynchronized manner so each part may, in parallel, be at a different stage of development (e.g., definition design, implementation, integration, qualification) at any given time. This approach, which attracted unsavory reputation, is under intensive scientific research and gaining due respect as a legitimate way to reduce elapsed time required to bring systems into the market.

ing component qualification before inclusion into the produced systems. VVT planning also includes developing an efficient production VVT strategy to assure good system components are delivered with a test subset that is viable and economical.

System Production Once the system is deemed ready, the next phase is to produce final products for sale or use. VVT activities include testing of purchased parts and the conduct of component qualification tests. VVT planning includes preparing to receive and process field failure data when the system is fielded.

System Use and Maintenance When regarding the overall system lifecycle one must also consider the VVT activities during the Use and Maintenance phase. The system is now fielded and under customer control. It operates in its intended environment and manned by operators who have been trained in its proper use. Maintenance should be performed in accordance with the policies and guidelines established during its development. Failures may occur due to component wear, operator error or unanticipated harsh environmental factors as well as defective design or poor manufacturing process. If these occur during the warranty period, the program/project team should have responsibility for correction and possibly additional rework if the failure has revealed a fundamental system deficiency. Also, during this phase, eventual improvements to the system functions are introduced, errors are eliminated, and systems are maintained.

System Disposal After the use of the system, its disposal becomes an important aspect which should have been planned from the earliest days of the system development. During this phase systems must be dismantled, recycled, if necessary, and/or finally disposed of.

In general, VVT activities are performed within this phase only for systems with public safety issues associated with the system disposal or for systems that had specific disposal-related requirements imposed during their development. In these cases, there are likely to be enabling technologies required (such as nuclear waste disposal) which will have VVT activities. If the program is of sufficiently long duration, the disposal-enabling technologies may require certification or validation that should be planned for in advance and executed when needed.

1.3.3 Views of the System

During the entire lifecycle, from system definition to system disposal, there are different views one could have on the system. Naturally, the most important view for this book is the "VVT" view, which focuses on all activities that are implemented to assure the required quality by means of verification, validation, and testing of the system or system components. Such activities should be performed during every lifecycle phase to assure the quality of intermediate or final lifecycle products. Beside this view, there are of course other views,

such as system management, systems engineering and configuration management, which are related but of secondary importance for this book.

System Management View System management includes activities concerned with organizational issues associated with a system or a product. These include:

- The subdivision of the development and production process into phases and activities
- The division and definition of the work to be done
- The regulation of communication
- The organization and control of the work flow

The activities set out in system management comprise planning and controlling of various activities, the allocation of internal roles and the setting up of an interface to units outside the project (i.e., subcontractors, management, etc.). Typically, system management contains the following main tasks: project initialization, detailed planning, project control, reporting, cost–benefit analysis, phase reviews, risk management, resource management, contractor management and training.

System Engineering View System engineering is that set of activities which directly leads to the development, production, use and maintenance and finally disposal of a system, as opposed to other activities related to system management, quality assurance and configuration management, which (crucial though they are) play a supporting role from the perspective of system construction. The system development lifecycle covers the following main activities:

- System requirement analysis
- Software/hardware requirement analysis
- System and subsystem design
- Component and subsystem implementation (hardware/software units)
- System integration
- System qualification

In system development, all activities directly relevant to the system development lifecycle process and the respective documents are grouped together. A system development lifecycle encompasses the complete set of activities that generate and implement engineering decisions about a system:

- What it should do (and not do)
- Which technologies should be used and where
- How it should be structured into parts
- How parts should be obtained (design-and-build, reuse-and-adapt, acquire, etc.)

- How VVT should be done
- How integration should be performed
- How to produce systems (for mass market or a small number of products)
- How to maintain systems and dispose of obsolete ones

Verification, Validation, and Testing View Conventional wisdom says that to produce competitive products one must identify the requirements and proceed to meet these in an efficient and effective way. This is a quality assurance process, which can be separated into three different levels: the organizational level, the process level and the product level. The activities relevant to the VVT view serve as the basis for the detailed explanation of activities and methods in the following chapters of this book.

Configuration Management View Configuration Management (CM) comprises those activities that must be performed in order to manage all the parts and their relationships and to support systems engineers in maintaining the integrity of the system. It is a service function that allows the various participants involved in the system engineering process to perform their perspective role confidently.

1.3.4 VVT Aspects of the System

Each individual activity describes one block of work of the project's complex network of tasks. Each VVT activity may be assigned to one of the following VVT aspects:

- *Prepare VVT Products.* This VVT aspect encompasses VVT activities related to preparation of VVT products, such as developing a certain VVT plan and designing and fabricating certain VVT tools or simulations.
- *Perform VVT Activities.* This VVT aspect encompasses VVT activities related to actual VVT of various system engineering products, for example, verifying a system design document and testing a package of software.
- *Participate in Reviews.* This VVT aspect encompasses VVT activities related to either participating in or conducting a system review, for example, participating in a system Preliminary Design Review (PDR) and conducting a Test Readiness Review (TRR).

1.4 METHODOLOGY APPLICATION

1.4.1 Introduction

In this section we begin to get to the heart of the subject matter in this book. VVT has developed over the years into a set of tools that are tried and proven to save time and money and ensure success in the design and building of complex systems. Having covered the preliminaries in the previous sections,

we concentrate here on the tools and techniques available for system VVT. We begin with an overview of the VVT methodology. The basis of this methodology is a process model that assists VVT planning by providing calculation of the cost and risk associated with the various VVT strategies. This process is a guide to modern VVT planning as performed by VVT practitioners, in coordination with the other stakeholders of the engineered system. As mentioned, a good VVT process does not "just happen." It is the product of thorough planning and strategy.

Since there is no such thing as a "typical" engineered system, what is good for one system in the way of VVT may not be good for another. So, we go on to show how VVT can be tailored to different kinds of systems, different organizations and different project parameters. Heuristics are described for tailoring VVT concepts to specific engineered systems based on project size/complexity and type (i.e., system or industry). Specific attention is paid to the electronics/avionics, aerospace, automotive, food packaging and steel production industries as representative of many other industries. Hints are given for ameliorating project risks by tailoring VVT.

An important issue is the means by which VVT can be monitored and stakeholders can be assured that VVT is properly applied. Remember the old adage, "The job is not complete until the paperwork is done." Of course, today paperwork does not necessarily imply the generation of paper documents. But, records do have to be kept and a trace of VVT steps and functions must be made. This is the only way to assure that the process works and that monies allocated for VVT have been properly spent. Among the necessary documents are the Project Management Plan (PMP), the Systems Engineering Management Plan (SEMP), the VVT Master Plan (VVT-MP), the Testability Program Plan (TPP), the Maintainability Program Plan (MPP), the Reliability Program Plan (RPP), the System Test Plan (SysTP), the Software Test Plan (STP, if appropriate), the First Article Inspection Plan (FAIP), the Production Plan (PP), the Maintenance Plan (MP), the Integrated Logistic Support Plan (ILSP) and the Disposal Plan (DP). While, for any specific system not all of these plans may be required, we provide fair details of what these documents consist. In summary, reading this section sets the stage for the following chapters, which cover the "how to" for implementing VVT.

1.4.2 VVT Methodology Overview

The basis of the VVT methodology is to apply an informed strategy and planning process to the selection and sizing of VVT activities. Through such a process, VVT activities, methods, tools and products are optimized to reduce project risk while improving cost, quality and development time. This book describes a process model that assists VVT planning by providing calculation of cost and risk associated with various VVT strategies. The effort required for performing the VVT strategy, planning, and modeling should be commensurate with the size of the project, so that the effort expended will be repaid in improved quality and reduced project cost, risk and development time.

Methodology for VVT Strategy and Planning The generic VVT process is depicted in Figure 1.10 (Lévárdy et al., 2004). It is an iterative process that can be applied to the entire system lifecycle, to a subset of the system lifecycle (e.g., system development) or to any of the individual lifecycle phases. The VVT process has four main segments: (1) VVT tailoring at the organization and project level, (2) Rough VVT planning at the system level, (3) Detailed VVT planning and (4) VVT execution.

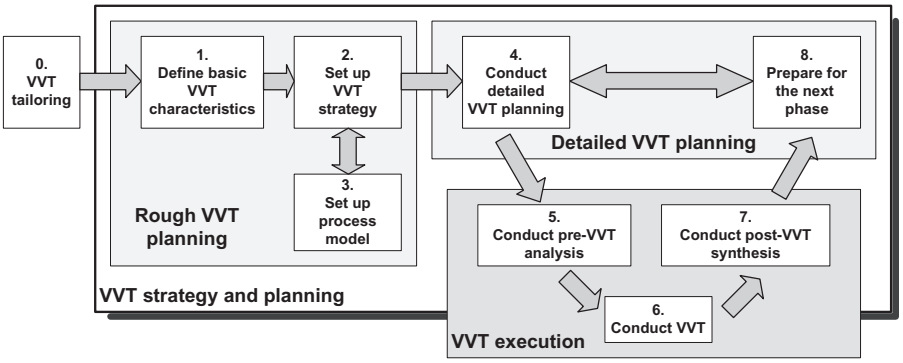

Figure 1.10 VVT methodology for strategy and planning (Lévárdy et al., 2004).

The VVT for strategy and planning encompass the following steps:

1. *VVT Tailoring.* Before starting a project, those managing the project should determine the factors that characterize the project and enterprise. Based on these factors, the project managers should tailor the VVT methodology to suit the project. Tailoring consists of high-level decisions about the use of this methodology and its parts based on knowledge of the organization and insights gained in earlier project.
2. *Rough VVT Planning.* At the outset of each project, it is necessary to plan the VVT process, at least in a rough manner, and establish a VVT strategy. The VVT strategy considers business objectives and their relationship to the project as well as issues related to programmatic and strategy risks. Strategy consists of creating a set of requirements and constraints that guide the VVT planning along with primary decisions about the VVT activities to follow. VVT rough planning uses the following three process groups:
   - *Define basic VVT characteristics.* This determines the basic characteristics that guide and bound the VVT strategy.
   - *Set up VVT strategy.* This codifies the strategy into a selection of activities and methods while also defining the requirement verification methods to be used.

- *Set up a VVT process model.* This uses the VVT process model to support the strategy definition by using calculation of cost, time and risk to explore alternative strategies.

3. *Detailed VVT Planning.* Throughout the system's lifecycle and especially at the beginning of each lifecycle phase, VVT engineers should reexamine or/and establish a detailed VVT plan. This plan should identify specific activities, methods, tools and products that will implement the actual VVT process. The VVT plan also identifies the types, formality and amount of effort to be applied to each VVT activity.
4. *VVT Execution.* The VVT execution process for each lifecycle phase will usually incorporate the following three process groups:
   - *Conduct a pre-VVT analysis.* This analysis will update the VVT strategy to incorporate changes as needed.
   - *Conduct VVT.* This is the actual execution of the VVT process for the relevant lifecycle phase.
   - *Conduct a post-VVT synthesis.* This analysis will update the future VVT strategy to incorporate anticipated changes as needed.

Importance of VVT Strategy and Planning A vital and effective VVT process enhances the technical success of a development program. A well-planned VVT strategy reduces program risk, whereas lack of adequate VVT planning can contribute to programmatic risks. Program costs are minimized when redundant testing is reduced or eliminated. Good VVT planning helps to eliminate redundant testing. Lowest risk is ensured when program strategy includes VVT at an early point in the program and provides continuous attention to VVT-related details. Figure 1.11 illustrates the areas where the implementation of the VVT methodology tends to improve the traditional company VVT processes.

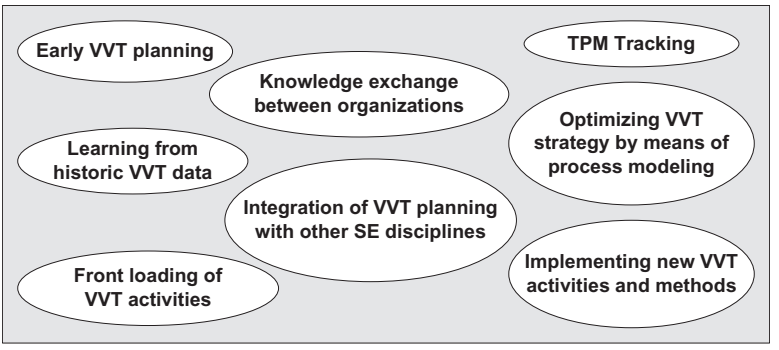

Figure 1.11 Key areas improved by using the VVT methodology (Lévárdy et al., 2004).

Philosophy for VVT Strategy and Planning A good VVT process does not just happen. It is the product of thorough planning and strategy. The philosophy driving VVT should be "Verify early, validate continuously." VVT must combine programmatic thinking with technical thinking. Ultimately, project success is determined in large measure by the effectiveness of its VVT. Technical success depends upon meeting or exceeding performance requirements. Good VVT supports both. A well-planned VVT will:

- Save money through reduced or eliminated test redundancy
- Protect the schedule by being efficient in demands for resources and time
- Assure technical success by identifying areas of performance risk
- Facilitate the Integration phase by ensuring robust component and subsystem interfaces
- Guarantee stakeholder delight by validating requirements against true needs early enough to effect timely change if needed

1.4.3 VVT Tailoring

The VVT methodology is intended to apply to a broad range of projects and enterprises. This section provides guidance and heuristic suggestions on how the unique factors of each project and enterprise may modify the strategy and planning process. Tailoring should be performed at two different levels:

- *VVT Tailoring for Each Organization/Industry.* This tailoring is usually performed once for the enterprise, with occasional updates. In addition, it can be performed on an organizational level for different product lines, thus establishing tailored VVT methodology for each product line. In the event a business undergoes major organizational changes, there might be a need to perform the tailoring again.
- *VVT Tailoring for Specific Projects.* This tailoring is usually performed at the beginning of each project or major replan as part of the VVT planning process.

Tailoring Parameters Three groups of tailoring parameters have been identified for tailoring the VVT methodology: (1) organization/project parameters, (2) programmatic risks and (3) product characteristics.

1. *Organization/Project Parameters.* Table 1.14 identifies three typical major organization and project parameters. These parameters are key discriminators between diverse organizations and product lines as well as projects and are used for both organizational and project VVT tailoring.

TABLE 1.14 Typical Organization/Project Parameters

Parameter	Characteristics
Project size	• Large—Multiteam projects usually more than several million dollars and more than one year duration • Small—Few staff members, limited budget (less than $1 million), few month schedule (less than one year)
Project complexity	• High—Involves many diverse entities or high projects requirements (e.g., performance requirements, aggressive schedule) • Low—Typically simple products manufactured in large quantities
Project type	• Concept exploration—Typically research projects • Technology demonstration—New concept/technology realization in a prototype (possibly limited) for customers' demonstration • Full-scale development/manufacturing—New product development and manufacturing • Maintenance—Improving existing products by fixing deficiencies or adding limited capabilities • Upgrade—Substantially improving existing products by introducing new capabilities

2. *Programmatic Risk Parameters.* Table 1.15 presents three typical programmatic risks that significantly affect VVT project tailoring and planning.

TABLE 1.15 Typical Programmatic Risk Parameters

Parameter	Characteristics
Unachievable schedule	Allocated time to completion is too short to deliver all required capabilities with required quality and maturity.
Insufficient budget	Allocated budget is too small to deliver all required capabilities with required quality and maturity.
Insufficient quality	Allocated resources (e.g., people, schedule, budget, facilities) are not sufficient to meet product quality requirements.

3. *Product Characteristic Parameters.* Table 1.16 presents six product characteristics affecting VVT activities, methods and tool selection.

TABLE 1.16 Typical Product Characteristic Parameters

Parameter	Characteristics
Critical	Mission-critical or safety/health-critical systems parts—Failure in these parts can cause significant human/financial/environmental damage.
Complex	Contains complex system requirements, architecture, real time, deployment, use, production or disposal. Complex systems can be defined as disproportionably large, intricate or convoluted.
Innovative	New technology/feature/capability that has not been previously proved and validated.
Changed	Existing system capability that must undergo limited upgrade/improvement.
Precise	Systems require meeting high-performance or precision requirements.
Need certification	System which requires formal approval/certification by regulatory agencies [e.g., Food and Drug Administration (FDA) and Federal Aviation Administration (FAA)]

Tailoring Heuristics: General Tailoring should always be done within a context and with the benefit of experience. While creating the VVT methodology, certain heuristics were identified. This section contains tailoring heuristics for each relevant parameter.

1. *Organization/Project Parameters.* Table 1.17 presents tailoring heuristics for project size/complexity.

TABLE 1.17 Heuristics for Tailoring Based on Project Size/Complexity

Parameter	VVT Heuristics
Large	• Use incremental or evolutionary VVT lifecycle. • Define detailed VVT process and schedule. • Use frequent informal and formal technical reviews. • Plan for concurrent and early integration activities. • Use formal detailed technical and management VVT documentation. • Use formal requirements and change control. • Adopt the following VVT methods: classification tree method, evolutionary testing, requirements tracing, hierarchical testing, defect tracing, regression testing, etc. • Automate VVT as much as practical. • Use high-end VVT tools and facilities.
Small	• Use less formal VVT process. • Consider merging VVT phases. • Use less formal reviews. • Focus on less formal and less detailed technical documentation. • Adopt VVT methods such as walkthrough.

2. *Project Type.* Table 1.18 presents tailoring heuristics for project type.

TABLE 1.18 Heuristics for Tailoring Based on Project Type

Parameter	VVT Heuristics
Concept exploration	• Use evolutionary VVT lifecycles. • Use less formal VVT process. • Use informal reviews. • Adopt the following VVT methods: simulation, model checking, benchmarking, etc.
Technology demonstration	• Use less formal VVT process. • Use less formal reviews. • Adopt the following VVT methods: prototyping, simulation, model checking, benchmarking.
Full-scale development/ manufacturing	• Use incremental or evolutionary VVT lifecycles. • Define detailed VVT process and schedule. • Use frequent informal and formal technical reviews. • Plan for concurrent and early integration activities. • Use formal detailed technical and management VVT documentation. • Use formal requirements and change control. • Adopt the following VVT methods: classification tree method, evolutionary testing, requirements tracing, hierarchical testing, defect tracing, regression testing, etc. • Automate VVT as much as practical. • Use high-end VVT tools and facilities.
Maintenance	Use regression testing, impact analysis, inspection and walkthrough.
Upgrade	Use regression testing, impact analysis, inspection and walkthrough.

3. *Industry Type.* Tables 1.19–1.22 present additional VVT tailoring characteristics and heuristics unique for each of the industry types examined in the SysTest project.

TABLE 1.19 Heuristics for Tailoring in Aerospace/Avionics Industry

- Mostly large projects evolving from previous or existing systems.
- Often projects involve large and critical systems of systems that require different tailoring for different subsystems.
- Mostly few-of-a-kind projects. Production is often in a few or tens of units (emphasizing development rather than production)
- Due to each customer's unique requirements, tailoring is required for essentially every project.
- Certification authorities are major VVT stakeholders.
- Real-life tests are generally mandatory.
- Many projects have aggressive schedule objectives leading to concurrent VVT and incremental lifecycles.
- Some customers require the transfer of technology and future support knowhow to their organizations. This implies delivering many enabling products to the customer and therefore requires their higher quality and increased VVT effort.
- Technology development projects require evolutionary lifecycles, prototyping, simulation, and Design Of Experiments (DOE) methods.
- Very long lifecycle (more than 30 years life span is not uncommon)

TABLE 1.20 Heuristics for Tailoring in Automotive Industry

- Production volumes vary between a few hundred cars in the top luxury segment to several hundred thousand in the economy class.
- Typical development cost for a new model lies between $100 million and $1 billion.
- New developments are usually introduced in the luxury car sector (because of cost as well as lower production volumes).
- Most automotive embedded systems are large distributed systems running on many central processing units (CPUs) and communicating via buses.
- Most projects impose hard time-to-market constraints resulting in aggressive schedules leading to concurrent VVT.
- High competition with other automobile manufacturers.
- Most projects involve a large number of subcontractors for the implementation of different components, e.g., software modules. This often implies close interaction with external processes and organizations.
- Worldwide distribution of products results in different components and subcontractors for different regions and in a widespread distribution of enabling products.
- Generally high-quality requirements.
- End-user/consumer products resulting in high usability requirements and corresponding VVT activities such as early simulations

TABLE 1.21 Heuristics for Tailoring in Food Packaging Industry

- Standard small–medium size product developments are based on previous knowledge, historical database and best practices.
- Standardized projects require tailoring only for the specific issued product properties. The other requirements must be comparable with the historical data.
- Large, complex and innovative equipment developments require particular attention to concept development and screening based on objective measurements.
- All products are human health critical. A set of procedural VVT activities must be applied in order to fulfill food production regulations.
- Large-scale tailoring is required only for innovative products.
- New products start with a technology demonstrations phase. This phase must be objectively assessed using appropriate metrics.
- Continuous VVT monitoring approach is essential for the final customer and the human health safety.
- Physical testing, particularly in the intended environment, is important but entails great expenditures. VVT tailoring may be appropriate in certain cases.

TABLE 1.22 Heuristics for Tailoring in Steel Production Industry

Steel production is a process of making steel slabs from iron ore. This industry presents several VVT tailoring characteristics:

- Massive production (e.g., 250.000 tons/year) with a few product critical parameters to be verified (e.g., weight and size of steel slabs as well as physical and chemical composition).
- Intensive production and speed rates that require production line monitoring and optimization.
- In general, faulty steel products can be corrected.
- Steel production lines are similar systems; therefore, VVT tailoring requirements are basically the same for most projects.

Tailoring Heuristics: Programmatic Risks This section contains some tailoring heuristics for ameliorating project risks (Table 1.23).

TABLE 1.23 Heuristics for Tailoring Based on Anticipated Project Risks

Risk	VVT Heuristics
Unrealistic schedule	• Negotiate the scope of VVT effort to reduce it to a realistic level. • Negotiate with the customer for a realistic schedule. • Use formal requirements/change control to avoid unauthorized scope increase. • Move some of the desired functionality into future versions. • Deliver the product in stages so VVT activities could be stretched over a longer period. • Use incremental VVT lifecycles. • Adapt less formal VVT process (less documentation, reviews, etc.). • Negotiate the quality of some parts—implement them to "just enough" quality degree, and not more. • Use testing facility in two or three shifts. • Get another testing facility and team for parallel testing in two facilities. • Start testing earlier with less mature subsystems.
Insufficient budget	• Use strict requirements/change control to avoid unbudgeted scope increase. • Negotiate the scope of VVT effort in order to reduce it. • Convince the customer to extend the schedule. • Transfer budget from less critical projects to a more critical project. • Negotiate acceptable quality. Identify ways to reduce VVT efforts spent on less critical requirements. • Adapt less formal VVT process (e.g., less documentation, reviews). • Start VVT with mature work products. • Conduct upstream requirements and design reviews (when it is least expensive to introduce change).
Insufficient quality	• Plan for increased VVT effort, schedule and budget. • Define Detailed VVT process. • Use domain experts for VVT of complex, risky or critical parts of the system. • Use frequent informal and formal technical reviews. • Build consensus about acceptable quality. • Adopt the following VVT methods: inspection, walkthrough, boundary value analysis, robustness testing, behavior testing, back-to-back testing, prototyping, etc. • Use high-end VVT tools and facilities.

Tailoring Heuristics: Product Characteristics This section contains some tailoring heuristics to accommodate product characteristics (Table 1.24).

TABLE 1.24 Heuristics for Tailoring Based on Product Characteristics

Characteristic	VVT Heuristics
Critical	• Perform criticality analysis and allocate more VVT effort for critical parts. • Conduct upstream requirements and design reviews, inspections and walkthroughs. • Use Independent Verification and Validation (IV&V) team. • Use hierarchical testing with caution not to leave out important tests. • Test enabling products more rigorously. • Adopt the following VVT methods: robustness testing, safety testing, model checking, boundary value analysis, Failure Modes and Effects Analysis (FMEA), etc. • Use high-fidelity models and simulations. • Use VVT automated tools to assure engineering data consistency.
Complex	• Use domain experts for VVT of complex parts. • Use formal inspections for requirements and design. • Use Model Checking, Simulations, and Back-to-back testing. • Emphasize interface VVT. • Use VVT automated tools to assure engineering data consistency.
Innovative	• Use evolutionary VVT lifecycle. • Emphasize validation activities with stakeholders. • Adopt the following VVT methods: prototyping, simulation, model checking and exploratory testing.
Changed	• Use waterfall VVT lifecycle strategy • Adopt the following VVT methods: regression testing and impact analysis.
Precise	• Test enabling products more rigorously • Adopt the following VVT methods: benchmarking, simulation and model checking.
Need certification	• Often certification requirements are not identified explicitly. The VVT cost and time required are very high and must be taken into account. • Employ regulatory domain experts.

1.4.4 VVT Documents

This section provides an overview of various strategy and planning documents that can be used in conjunction with the VVT methodology. In other words, these documents either are produced by VVT engineers or contain sections related to the VVT process. Documents that control the definition of the

project from inception to conclusion should contain clear statements about the VVT strategy. The documents discussed below play specific roles in the project. Project management usually decides which documents are required for a specific project.

Project Management Plan (PMP)

1. *Review.* The PMP, which sometimes is identified as an Engineering Program Plan (EPP), identifies the activities, critical milestones and events in relationship to systems engineering management and schedule control and typically includes the following events as a minimum:

   - Formal technical review for the system(s), subsystem(s), and their corresponding configuration items
   - Trials and test releases (if applicable)
   - Engineering releases
   - Production release
   - Acceptance tests
   - Logistic support events
   - Formal audits
   - Formal progress reviews

   These data identify the major activities and events required by the Statement of Work (SOW) or similar contract document defining the scope of the work. Any planned program strategies and build planning are identified in detail appropriate to the information available. The project management plan contains the project schedule(s) and identifies the appropriate activities, showing when each activity is initiated, the availability of draft and final deliverables and other milestones, and the due date for the completion of each activity. In addition, entry and exit criteria should be defined for each activity, that is, the conditions that should exist for the activity to start and for the activity to stop.
2. *Plan Source Pointer.* IEEE 1058.1 provides guidance for software PMP preparation. While its utility for hardware-oriented or hybrid developments is not proven, it is nevertheless an excellent resource. It can be purchased from the IEEE.

   The European Cooperation for Space Standardization document ECSS-M-30A, Project Phasing and Planning, provides planning principles and guidance but no template for the plan itself. It is an initiative established to develop a set of user-friendly standards to be utilized in all European space activities. Another source of PMP templates is the DI-MGMT-80004 management plan and the older DI-A-5239B management plan, which was superseded by DI-MGMT-80004.

Systems Engineering Management Plan (SEMP)

1. *Overview.* The SEMP establishes the overall plan for the technical development of a specific project. The SEMP defines the system performance parameters and preferred system configuration to satisfy the technical requirements and provides the planning and control of technical program tasks. It includes integration of engineering specialties and management of the entire system development effort. This includes design engineering, computer software engineering, specialty engineering, test engineering, logistics engineering, quality evaluation, and production engineering. The ultimate objective of the SEMP is to provide a disciplined framework to meet cost, technical performance, and quality and schedule objectives for the project or program. It is important that the SEMP establish the VVT philosophy for the program.
2. *Plan Source Pointer.* There are several good sources for a model SEMP. The first is Appendix C of the INCOSE *Systems Engineering Handbook*. The second is from The European Cooperation for Space Standardization document ECSS-E-10, Part 1B, systems engineering (November 2004), Appendix A. Some online sources are available but are not always free to the public. For example, the military standard DI-MGMT-81024 System Engineering Management Plan (SEMP). Two older standards that provide useful templates are the Data Item Description DI-S-3618, System Engineering Management Plan (SEMP), and DI-E-7144, Simulator System Engineering Management Plan (SEMP), both of which were superseded by DI-MGMT-81024.

Test and Evaluation Management Plan (TEMP)

1. *Overview.* The TEMP defines the approach to test and evaluate the project from both a technical and a management perspective. The TEMP defines the system test program and preferred test infrastructure necessary to satisfy the VVT philosophy set forth in the SEMP and meets the verification requirements. The TEMP provides for the planning and control of test program tasks.
2. *Plan Source Pointer.* The TEMP is similar in concept to the SEMP in that it provides an overall plan for the development of the testing program for the project. It can follow the organization of the SEMP. Another source of document structure is the U.S. military specification Data Item Descriptions (DID). One, which could fulfill the needs of the TEMP, is DI-NDTI-81284, Test and Evaluation Program Plan (TEPP).

Verification Validation and Testing Master Plan (VVT-MP)

1. The Test and Evaluation (TEMP) issued by the U.S. DoD was designed to manage and plan system testing (in the narrow sense of the term) during the system qualification phase. It does not deal with the multitude of VVT activities which are nontesting by nature or occurring at other

system lifecycle phases. A proposed VVT-MP which deals with the strategic planning of the entire VVT process in a broader manner is provided in Appendix B.

Testability Program Plan (TPP)

1. *Overview.* The TPP identifies the performing activity approach for implementing a testability program. It is mostly used to provide the acquirer with a basis for review and evaluation of the testability program. It usually is applicable for all systems and equipment development programs.
2. *Plan Source Pointer.* The TPP should be prepared in accordance with MIL-HDBK-2165, *Testability Handbook for Systems and Equipment*. Data item description and documentation guidance can be found in DI-MNTY-81604, Maintainability/Testability Demonstration Test Plan.

System Test Plan (SysTP)

1. *Overview.* The SysTP elucidates how to implement a system testing program. The purpose of the SysTP is to assure attainment of the requirements of the acquisition as stated in the system/subsystem specification.

   Requirement compliance may be proven through one of five methods, that is, analysis, inspection, demonstration, testing or certification. The SysTP describes the approach to using all five methods throughout the program life in a coordinated and efficient fashion. The SysTP considers resource allocation, facilities planning and overall scheduling of test activities as they support the overall project schedule.
2. *Plan Source Pointer.* See Section 2.6.1 on how to generate a qualification/acceptance SysTP.

Software Test Plan (STP)

1. *Overview.* The STP identifies the performing activity approach for implementing an organized software verification program. The purpose of the STP is to assure attainment of the requirements of the software system as stated in the System/Subsystem Specification. Requirement compliance may be proven at different levels during the software development process. Requirements proven through an instrumented "test" at a module or unit level may be verified using a demonstration of performance at higher levels. The STP describes the approach to use the appropriate verification methods (analysis, inspection, demonstration, testing or certification) throughout the software development in a coordinated and efficient fashion. The STP considers resource allocation, facilities planning and overall scheduling of test activities as they support the overall software development and integration schedule.

2. *Plan Source Pointer.* The STP structure should follow the software development approach. Object-oriented software is tested and integrated differently than modular or functional software implementations. Military standards templates appropriate for STP documentation are DI-IPSC-81438A, Software Development and Documentation, and the family of documents it superseded—DI-NDTI-80808, Test Plans/Procedures; DI-MCCR-80307, Software General Unit Test Plan; DI-MCCR-80308, Software System Integration and Test Plan; and DI-MCCR-80309, Software System Development Test and Evaluation Plan—all of which provide templates for STP. The legacy DIDs may be found to be useful with software projects using modular, functional code architectures. The now-superseded MIL-STD-498, Software Development and Documentation, had a well-organized software approach, which can be found in IEEE/EIA 12207, Standard for Software Lifecycle Processes.

Reliability Program Plan (RPP)

1. *Overview.* The RPP identifies the performing activity approach for implementing a reliability program. The purpose of the RPP is to assure attainment of the reliability requirements of the system as stated in the system/subsystem specification.

   Reliability should be stated initially in development specifications as a goal with a lower minimum acceptable requirement. In this case, realistic requirements are determined and incorporated later in the development specification together with the requirements for system demonstration. In general, both reliability and performance should be considered of similar importance, although this view may vary from one project to another.
2. *Plan Source Pointer.* The RPP should be prepared in accordance with MIL-STD-785. Additional details can be obtained using MIL-HDBK-781A, *Handbook for Reliability Test Methods, Plans, and Environments for Engineering, Development Qualification, and Production*.

Maintainability Program Plan (MPP)

1. *Overview.* The MPP identifies the performing activity approach for implementing a maintainability program to support the fielded system. The purpose of the MPP is to improve operational readiness, reduce maintenance manpower needs, reduce system lifecycle cost and provide data essential for management. In addition, the MPP should assure attainment of the maintenance requirements of the system as stated in the system/subsystem specifications. These usually include:
   - Time (e.g., turnaround time, time to repair, time between maintenance actions)

- Rate (e.g., maintenance hours per operating hours, frequency of preventative maintenance)
- Complexity (e.g., number of people and skill levels, variety of support equipment)

The expectation of carrying out repairs by substitution of components is also defined in the MPP.

2. *Plan Source Pointer.* An MPP should be prepared in accordance with the MIL-STD-470B. Additional guidance can be obtained from MIL-HDBK-2084, *Handbook for Maintainability of Avionic and Electronic Systems and Equipment*. Another resource for producing the maintenance plan is MIL-T-81821 (3), General Specification for Trainers, Maintenance, Equipment and Services.

First Article Inspection Plan (FAIP)

1. *Overview.* The FAIP identifies the performing activity approach for implementing first article inspection. The purpose of the FAIP is to fulfill Physical Configuration Audit (PCA) requirements of the acquisition as articulated in the SOW or other overarching program requirement documentation. The requirements are usually fulfilled by the drawings and supporting lists.
2. *Plan Source Pointer.* The FAIP can draw guidance from DI-QCIC-81110, Inspection and Test Plan, and either DI-NDTI-81307A, First Article Qualification Test Plan, or the older DI-T 5315, First Article Qualification Test Plan. Another source is the Society of Automotive Engineers, Standard SAE-AS9102A, Aerospace First Article Inspection Requirement.

Production Plan (PP)

1. *Overview.* The PP identifies the performing activity approach for implementing production of the system that is being developed and is being taken into a production phase. The PP defines the planning and control of production tasks. It includes integration between the production organization and engineering specialties and the management of an integrated effort. This includes design engineering, computer software engineering, specialty engineering, test engineering, logistics engineering, quality evaluation, and production engineering with the goal of improving production. The ultimate objective of the PP is to provide a disciplined framework to meet production cost and quality and schedule objectives for the system in a production environment. The PP should establish the VVT philosophy for production.
2. *Plan Source Pointer.* This plan should be written in accordance with the specific requirement of the project.

Integrated Logistic Support Plan (ILSP)

1. *Overview.* The ILSP identifies the approach the performing activity should take for implementing a logistic program to support the fielded system. The purpose of ILSP is to assure attainment of the logistic requirements of the system as stated in the system/subsystem specification in a manner that is integrated into all aspects of the program. This addresses the inclusion of design features, which facilitates logistic support, including maintenance, transportation and repair.
2. *Plan Source Pointer.* The European Cooperation for Space Standardization document ECSS-M-70A 19 (April 1996), Integrated L ogistic Support, provides general information and guidance of integrated logistic support and planning principles but no template for the plan itself. ECSS-M-70A 19 is available at the ECSS website (http://www.ecss.nl). Other online resources of this nature are available but are not free to the public. Military standards provide a broad spectrum of ILSP material to considerable depth if the investment is warranted. The U.S. Department of the Army standard DA PAM 700-50, Integrated Logistic Support: Developmental Supportability Test and Evaluation Guide, currently provides top-level guidance on ILSP.

Disposal Plan (DP)

1. *Overview.* The DP identifies the performing activity approach for disposing of the system. The purpose of the DP is to fulfill requirements of the acquisition with respect to an orderly and safe disposal of a system whose components or subsystems impose a public safety hazard or serious environmental threat. A DP is not ordinarily required in nondangerous procurements.
2. *Plan Source Pointer.* This plan should be written in accordance with the specific requirement of the project. The DP could be based on the DoD 4160.21-M, *Defense Materiel Disposition Manual*, dated August 18, 1997 (see http://www.dtic.mil/whs/directives/corres/html/416021m.htm).

1.5 REFERENCES

Addy, A. E., Verification and Validation in Software Product Line Engineering, Dissertation, Department of Computer Science and Electrical Engineering, College of Engineering and Mineral Resources, West Virginia University, 1999.

ANSI/ITAA EIA-632, Processes for Engineering a System, American National Standards Institute/Information Technology Association of America, Sept. 1, 2003.

Balci, O., Verification, Validation, and Accreditation, in *Proceedings of the 1998 Winter Simulation Conference*, Washington, DC, Dec. 13–16, Piscataway, NJ, 1998, pp. 41–48.

Balci, O., Ormsby, F. W., Carr, T. J., and Saadi, D. S., Planning for Verification, Validation, and Accreditation of Modeling and Simulation Applications, in *Proceeding of the 2000 Winter Simulation Conference*, Orlando, FL, Dec. 2000.

Bertalanffy, V. L., *General System Theory: Foundations, Development, Applications*, George Braziller. 1976.

Boehm, B., Software Defects Reduction Top 10 List, *IEEE Computer*, 34(1), Jan. 2001.

Braha, D., Minai, A. A., and Bar-Yam, Y. (Eds.), *Complex Engineered Systems: Science Meets Technology*, Springer, 2006.

Browning, R. T., Modeling and Analyzing Cost, Schedule, and Performance in Complex Systems Product Development, Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, 1998.

Browning, R. T., Sources of Performance Risk in Complex Systems Development, paper presented at INCOSE1999, Brighten England, June 1999.

Capers, J., *Applied Software Measurement: Assuring Productivity and Quality*, Mcgraw-Hill, New York, 1996.

DA PAM 70050;DA PAM 700-50, Integrated Logistic Support: Developmental Supportability Test and Evaluation, Department of the Army, Washington, DC.

DI-E-7144, Data Item Description, System Engineering Management Plan (SEMP), superseded by DI-MGMT-81024, June 1984.

DI-IPSC-81438A, Data Item Description, Software Test Plan (STP), Dec. 1999.

DI-MCCR-80307, Data Item Description, Software General Unit Test Plan (STP).

DI-MCCR-80308, Data Item Description, Integration and Test Plan.

DI-MCCR-80309, Data Item Description, Development Test and Evaluation Plan.

DI-MGMT-81024, Data Item Description, System Engineering Management Plan (SEMP), Aug. 1990.

DI-MNTY-81604, Data Item Description, Maintainability/Testability Demonstration Test Plan, Feb. 2001.

DI-NDTI-80808, Data Item Description, Test Plans/Procedures, May 1989.

DI-NDTI-81284, Data Item Description, Test and Evaluation Program Plan (TEPP), Sept. 1992.

DI-NDTI-81307A, Data Item Description, First Article Qualification Test Plan and Procedures, Nov. 2006.

DI-QCIC-81110, Data Item Description, Inspection and Test Plan, Dec. 1990.

DI-S-3618, Data Item Description, Systems Engineering Management Plan (SEMP), U.S. Department of Defense, Feb. 1970.

DI-T-5315, Data Item Description, First Article Qualification Test Plan, U.S. Department of Defense.

DoD 4160.21-M, Defense Materiel Disposition Manual, U.S. Department of Defense, Washington, DC, Aug. 1997.

DoD 5000.59, Modeling and Simulation (M&S) Management, Department of Defense Directive, August 2007.

ECSS-E-10, Part 1B, *European Cooperation for Space Standardization*, System Engineering branch, Nov. 2004.

ECSS-M-70A, Integrated Logistic Support, *European Cooperation for Space Standardization*, Apr. 1996.

Engel, A., et al., Developing Methodology for Advanced Systems Testing—SYSTEST, research grant proposal for the European Commission, Research Proposal Office, GRD1-2001-40487, May 2001.

Fairley, E. R., *Software Engineering Concepts*, McGraw Hill, New York, 1985.

Fente, J., Knutson, K., and Schexnayder, C., Defining a Beta Distribution Function for Construction Simulation, in *Proceedings of the 1999 Winter Simulation Conference*, Vol. 2, Squaw Peak Resort, Phoenix, AZ, Dec. 1999, pp. 1010–1015.

Gonzalez, A., and Barr, V., Validation and Verification of Intelligent Systems—What Are They and How Are They Different? *J. Exper. Theor. Artif. Intell.*, 12(4), Oct. 2000.

Haimes, Y. Y., *Risk Modeling, Assessment, and Management*, Wiley-Interscience, New York, 1998.

Haimes, Y. Y., Kaplan, S., and Lambert, J. H., Risk Filtering, Ranking, and Management Framework Using Hierarchical Holographic Modeling, *Risk Anal.*, 22(2), 383–398, 2002.

IEEE 6101991IEEE 610-1991, IEEE Computer Dictionary—Compilation of IEEE Standard Computer Glossaries, Institute of Electrical and Electronics Engineers, New York, 1991.

IEEE/EIA 12207IEEE/EIA 12207, Standard for Software Lifecycles Processes, Institute of Electrical and Electronics Engineers/Electronic Industries Association, 1996.

INCOSE-TP-2003-002-03.1, C. Haskins (Ed.), *Systems Engineering Handbook—A Guide for System Lifecycles Processes and Activities*, Version 3.1, INCOSE, Aug. 2007.

ISO/IEC 15288ISO/IEC 15288, Systems and Software Engineering—System Lifecycles Processes, International Organization for Standardization/International Electrotechnical Commission, 2008.

ISO/IEC 15288ISO/IEC 15288, Systems and Software Engineering—System Lifecycles Processes, International Organization for Standardization/International Electrotechnical Commission, 2008.

Juran, J. M., and Gryna, F. M., *Quality Planning and Analysis: From Product Development Through Use*, 2nd ed., McGraw-Hill, New York, 1980.

Lake, J., *V & V in Plain English*, INCOSE, Brighton, UK, June 1999.

Lamm, A. G., and Haimes, Y. Y., Assessing and Managing Risks to Information Assurance: A Methodological Approach, *Syst. Eng. J.*, 5(4), 286–314, Nov. 2002.

Lévárdy, V., Hoppe, M., and Honour, E., Verification, Validation & Testing Strategy and Planning Procedure, in *Proceedings of the 14th Annual International Symposium of INCOSE*, Toulouse, France, June 20–24, 2004.

Martin, N. J., and Bahill, A. T., *Systems Engineering Guidebook: A Process for Developing Systems and Products*, CRC Press, Boca Raton, FL, 1996.

Millard, R. L., Value stream analysis and mapping for product development, Master's thesis in Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA, June 2001.

MIL-HDBK-781A, *Handbook for Reliability Test Methods, Plans, and Environments for Engineering, Development, Qualification, and Production*, Revision A.

MIL-HDBK-2084, *Handbook for Maintainability of Avionic and Electronic Systems and Equipment*, July 1995.

MIL-HDBK-2165, *Testability Handbook for Systems and Equipment, Naval Sea Systems Command*, July 1995.

MIL-STD-470B, Maintainability Program for Systems and Equipment, May 1989.

MIL-STD-498, Software Development and Documentation, Dec. 1994.

MIL-STD-785-Rev B, Reliability Program for Systems and Equipment, Sept. 1980.

MIL-STD-882c, System Safety Program Requirements, U.S. Department of Defense, Jan. 19, 1993.

MIL-T-81821 (3), Trainers, Maintenance, Equipment and Services General Specification, Mar. 1983.

Montgomery, C. D., *Introduction to Statistical Quality Control*, 4th ed., Wiley, New York, 2001.

Morgan, J. M., High performance product development: a systems approach to a lean product development process, Ph.D. thesis, University of Michigan, 2002.

Muessig, R. P., Laack, R. D., and Wrobleski, W. J., Optimizing the Selection of VV&A Activities—A Risk/Benefit Approach, paper presented at Winter Simulation Conference, Atlanta GA, Dec. 7–10, 1997, pp. 60–66.

Oppenheim, W. B., Lean Product Development Flow, *Syst. Eng.*, 7(4), 352–376, 2004.

Rechtin, E., *Systems Architecting*, Prentice-Hall, Englewood Cliffs, NJ, 1990.

SAE-AS9102A, Aerospace First Article Inspection Requirement, Society of Automotive Engineers, January, 2004.

Sörqvist, L., On Poor Quality Costing, Ph.D. Thesis, Department of Production Engineering, Royal Institute of Technology, Stockholm, Sweden, Mar. 1998.

Womack, P. J., and Jones, T. D., *Lean Thinking: Banish Waste and Create Wealth in Your Corporation*, Free Press; 2nd edition, 2003.