7

CONCEPT EXPLORATION

7.1 DEVELOPING THE SYSTEM REQUIREMENTS

Chapter 6 discussed the process of needs analysis, which is intended to provide a well-documented justification for initiating the development of a new system. The process also produces a set of operational requirements (or objectives) that describe what the new system must be designed to do. Assuming that those responsible for authorizing the initiation of a system development have been persuaded that these preliminary requirements are reasonable and attainable within the constraints imposed by time, money, and other external constraints, the conditions have been achieved for taking the next step in the development of a new system.

The principal objective of the concept exploration phase, as defined here, is to convert the operationally oriented view of the system derived in the needs analysis phase into an engineering-oriented view required in the concept definition and subsequent phases of development. This conversion is necessary to provide an explicit and quantifiable basis for selecting an acceptable functional and physical system concept, and then for guiding its evolution into a physical model of the system. It must be

Systems Engineering Principles and Practice, Second Edition. Alexander Kossiakoff, William N. Sweet, Samuel J. Seymour, and Steven M. Biemer

remembered, however, that the performance requirements are an interpretation, not a replacement of operational requirements.

As in the case of operational requirements, the derivation of system performance requirements must also simultaneously consider system concepts that could meet them. However, to ensure that the performance requirements are sufficiently broad to avoid unintentionally restricting the range of possible system configurations, it is necessary to conceive not one, but to explore a variety of candidate concepts.

New systems that strive for a major advance in capability over their predecessors, or depend on the realization of a technological advance, require a considerable amount of exploratory research and development (R & D) before a well-founded set of performance requirements can be established. The same is true for systems that operate in highly complex environments and whose characteristics are not fully understood. For these cases, an objective of the concept exploration phase is to acquire the needed knowledge through applied R & D. This objective may sometimes take several years to accomplish, and occasionally, these efforts prove that some of the initial operational objectives are impracticable to achieve and require major revision.

For the above reasons it, is appropriate that this chapter, which deals with the development of system requirements, is entitled "Concept Exploration." Its intent is to describe the typical activities that take place in this phase of system development and to explain their whys and hows.

The discussion that follows is generally applicable to all types of complex systems. For information systems, in which software performs virtually all the functionality, the section on software concept development in Chapter 11 discusses software system architecture and its design and should also be consulted.

Place of Concept Exploration Phase in the System Life Cycle

The place of the concept exploration phase in the overall system development process is shown in Figure 7.1. It is seen that the top-level system operational requirements

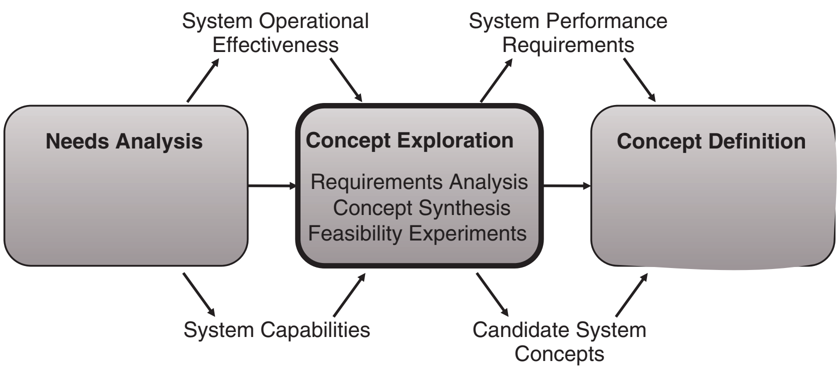

Figure 7.1. Concept exploration phase in a system life cycle.

come from needs analysis, which establishes that the needs are justified and that a development program is feasible within prescribed bounds. The outputs of the concept exploration phase are a set of system performance requirements down to the subsystem level and a number of potential system design concepts that analysis indicates to be capable of fulfilling those requirements.

While the formally defined concept exploration phase has a well-defined beginning and end, many of the supporting activities do not. For example, the exploratory development of advanced technological approaches or the quantitative characterization of complex system environments often begins before and extends beyond the formal terms of this phase, being supported by independent research and development (IRAD) or other nonproject funds. Additionally, considerable preliminary concept definition activity usually takes place well before the formal beginning of this phase.

The specific content of the concept exploration phase depends on many factors, particularly the relationship between the customer and the supplier or developer, and whether the development is needs driven or technology driven. If the system developer and supplier are different from the customer, as is frequently the case in needs-driven system developments, the concept exploration phase is conducted in part by the customer's own organization or with the assistance of a systems engineering agent engaged by the customer. The focus is on the development of performance requirements that accurately state the customer's needs in terms that one or more suppliers could respond to with specific product concepts. In the case of a technology-driven system development, the concept exploration phase is often conducted by the system developer and is focused on ensuring that all viable alternative courses of action are considered before deciding whether or not to pursue the development of a new system. In both cases, a primary objective is to derive a set of performance requirements that can serve as the basis of the projected system development and that have been demonstrated to ensure that the system product will meet a valid operational need.

For many acquisition programs, the period between the approval of a new system start and the availability of budgeted funds is often used to sponsor exploratory contractor efforts to advance technologies related to the anticipated system development.

System Materialization Status

The needs analysis phase was devoted to defining a valid set of operational objectives to be achieved by a new system, while a feasible system concept was visualized only as necessary to demonstrate that there was at least one possible way to meet the projected need. The term "visualize" is meant to connote the conceptualization of the general functions and physical embodiment of the subject in the case of needs analysis at the subsystem level.

Thus, in the concept exploration phase, one starts with a vision based generally on the above feasible concept. The degree of system materialization addressed in this phase has progressed to the next level, namely, the definition of the functions that the system and its subsystems must perform to achieve the operational objectives, and to the

TABLE 7.1. Status of System Materialization of the Concept Exploration Phase

	Phase					
	Concept development			*Engineering development*		
Level	Needs analysis	Concept exploration	Concept definition	Advanced development	Engineering design	Integration and evaluation
System	Define system capabilities and effectiveness	Identify, explore, and synthesize concepts	Define selected concept with specifications	Validate concept		Test and evaluate
Subsystem		Define requirements and ensure feasibility	Define functional and physical architecture	Validate subsystems		Integrate and test
Component			Allocate functions to components	Define specifications	Design and test	Integrate and test
Subcomponent		Visualize		Allocate functions to subcomponents	Design	
Part					Make or buy	

visualization of the system's component configuration, as illustrated in Table 7.1 (an overlay of Table 4.1).

Systems Engineering Method in Concept Exploration

The activities in the concept exploration phase and their interrelationships are the result of the application of the systems engineering method (see Chapter 4). A brief summary of these activities is listed below; the names of the four generic steps in the method are shown in parentheses.

Operational Requirements Analysis (Requirements Analysis). Typical activities include

- analyzing the stated operational requirements in terms of their objectives; and
- restating or amplifying, as required, to provide specificity, independence, and consistency among different objectives, to assure compatibility with other related systems, and to provide such other information as may be needed for completeness.

Performance Requirements Formulation (Functional Definition). Typical activities include

- translating operational requirements into system and subsystem functions and
- formulating the performance parameters required to meet the stated operational requirements.

Implementation Concept Exploration (Physical Definition). Typical activities include

- exploring a range of feasible implementation technologies and concepts offering a variety of potentially advantageous options,
- developing functional descriptions and identifying the associated system components for the most promising cases, and
- defining a necessary and sufficient set of performance characteristics reflecting the functions essential to meeting the system's operational requirements.

Performance Requirements Validation (Design Validation). Typical activities include

- conducting effectiveness analyses to define a set of performance requirements that accommodate the full range of desirable system concepts; and
- validating the conformity of these requirements with the stated operational objectives and refining the requirements if necessary.

The interrelationships among the activities in the above steps in the systems engineering method are depicted in the flow diagram of Figure 7.2.

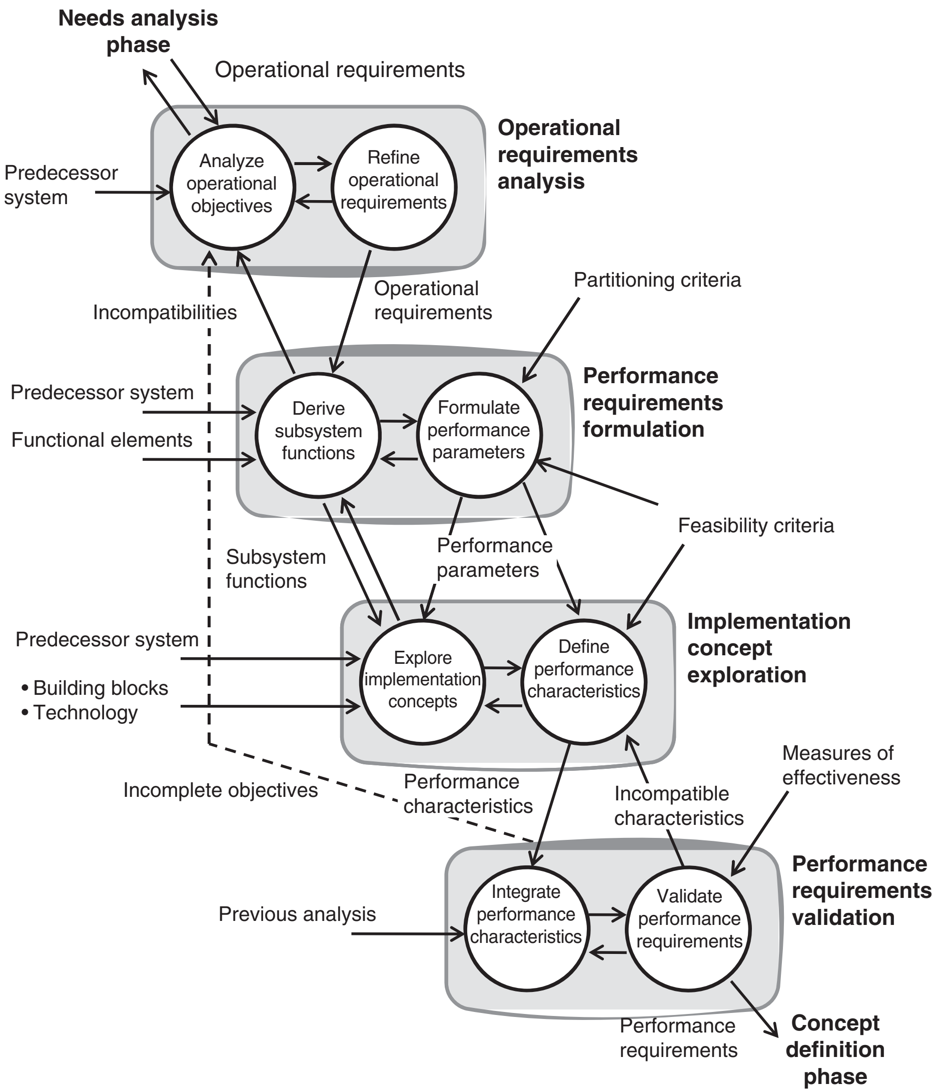

Figure 7.2. Concept exploration phase flow diagram.

7.2 OPERATIONAL REQUIREMENTS ANALYSIS

As in all phases of the system development process, the first task is to understand thoroughly, and, if necessary, to clarify and extend, the system requirements defined in the previous phase (in this case the operational requirements). In so doing, it is important to be alert for and to avoid shortcomings that are often present in the operational requirements as initially stated. We use a general process, known as *requirements*

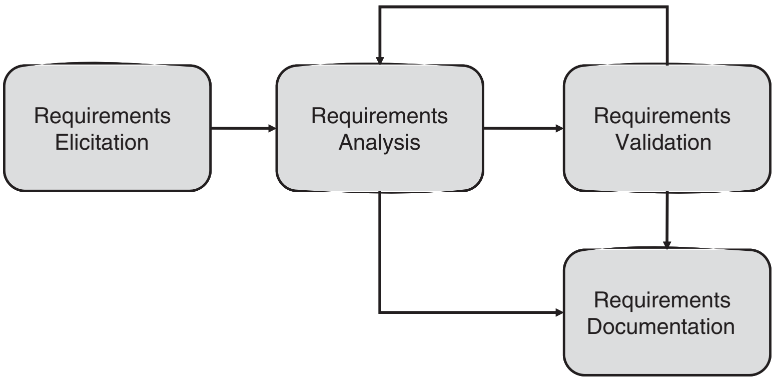

Figure 7.3. Simple requirements development process.

analysis, to identify and discover performance requirements, to synthesize and minimize initial sets of requirements, and finally, to validate the final set of requirements. This requirements analysis process, as mentioned in Chapter 4, occurs at each phase. However, the majority of effort occurs in the concept exploration phase where operational requirements are transformed into system performance requirements with measurable thresholds of performance. These system performance requirements tend to be the basis for contractual agreements between the customer and the developer and therefore need to be accurate and concise.

Figure 7.3 depicts the general process for developing requirements. Of course, this would be tailored to a specific application. The first activity involves the creation of a set of requirements. It is rare this occurs out of whole cloth—typically, a source of needs exists. In the concept exploration phase, a set of operational needs and requirements have been established. However, those needs and requirements are typically expressed in the language and context of an operator or user. These must be translated into a set of system-specific requirements describing its performance.

Requirements Elicitation

When analysts are developing operational requirements, they rely heavily on input from users and operators, typically through market surveys and interviews. When analysts are developing performance requirements, they rely on both people and studies. Initially, the customer (or buying agent within an organization) is able to provide thresholds of affordability and levels of performance that are desirable. But subject matter experts (SMEs) can also provide performance parameters as a function of technology levels, cost, and manufacturability. Previous studies and system development efforts can also assist in determining performance requirements. And finally, a requirements analyst performs a system effectiveness analysis to provide insight into the level of performance needed. All of these sources provide the analyst with an initial set of performance requirements.

Many times, however, this set of requirements contains inconsistencies, or even dichotomies. Further, many requirements are redundant, especially when they come from different sources. So the analyst must conduct a synthesis to transform an initial set of requirements into a concise, consistent set. More information is provided in Section 7.3 on formulating requirements.

A useful approach to developing requirements of any type is to ask the six interrogatives: who, what, where, why, when, and how. Of course, different types of requirements focus on different interrogatives, as described in Chapter 6. Operational requirements focus on the "why," defining the objectives and purpose of the system. Performance requirements focus on the "what," defining what the system should do (and how well).

Requirements Analysis

This activity starts with an initial set of requirements from the elicitation stage. Individual requirements as well as the set as a whole are analyzed for various attributes and characteristics. Some characteristics are desirable, such as "feasible" and "verifiable." Other characteristics are not, such as "vague" or "inconsistent."

For each requirement, a set of tests (or questions) is applied to determine whether the requirement is valid. And while many tests have been developed by numerous organizations, we present a set of tests that at least form a baseline. These tests are specific to the development of system performance requirements.

1. Is the requirement traceable to a user need or operational requirement?
2. Is the requirement redundant with any other requirement?
3. Is the requirement consistent with other requirements? (Requirements should not contradict each other or force the engineer to an infeasible solution.)
4. Is the requirement unambiguous and not subject to interpretation?
5. Is the requirement technologically feasible?
6. Is the requirement affordable?
7. Is the requirement verifiable?

If the answer to any of the questions above is "no," then the requirement needs to be revised, or possibly omitted. In addition, other requirements may need to be revised after performing this test.

In addition to individual requirements tests, a collective set of tests is also performed (usually after the individual tests have been performed on each requirement).

1. Does the set of requirements cover all of the user needs and operational requirements?
2. Is the set of requirements feasible in terms of cost, schedule, and technology?
3. Can the set of requirements be verified as a whole?

Both types of tests may need to be iterated before a final set of performance requirements exists.

Requirements Validation

Once a set of performance requirements is available, the set needs to be validated. This may be accomplished formally or informally. Formal validation means using an independent organization to apply various validation methods to validate the set of requirements against operational situations (i.e., scenarios and use cases) and to determine whether the requirements embodied within a system concept could achieve the user needs and objectives. Informal validation at this point means reviewing the set of requirements with the customer and/or users to determine the extent and comprehensiveness of the requirements. Section 7.5 provides further detail on the requirements validation process.

Requirements Documentation

A final, important activity is the documentation of the performance requirements. This is typically accomplished through the use of an automated tool, such as DOORS. Many tools exist that manage requirements, especially large, complex requirements hierarchies. As system complexity increases, the number and types of requirements tends to grow, and using simple spreadsheet software may not be sufficient to manage requirements databases.

Characteristics of Well-Stated Requirements

As mentioned above, the requirements analysis process leads to a concise set of performance requirements. This section examines the challenges associated specifically with translating operational requirements to performance requirements.

Since operational requirements are first formulated as a result of studies and analyses performed outside a formal project structure, they tend to be less complete and rigorously structured than requirements prepared in the subsequent managed phases of the development and are mainly oriented to justifying the initiation of a system development. Accordingly, in order to provide a valid basis for the definition of system performance requirements, their analysis must be particularly exacting and mindful of frequently encountered deficiencies, such as lack of specificity, dependence on a single assumed technical approach, incomplete operational constraints, lack of traceability to fundamental needs, and requirements not adequately prioritized. Each of these is briefly discussed in the succeeding paragraphs.

In an effort to cover all expected operating conditions (and to "sell" the project), operational requirements are often overly broad and vague where they should be specific. In the case of most complex systems, it is necessary to supplement the basic requirements with a set of well-defined operational scenarios that represent the range of conditions that the system is required to meet.

The opposite problem occurs if operational requirements are stated so as to be dependent on a specific assumed system configuration. To enable consideration of alternative system approaches, such requirements need to be restated to be independent of specific or "point" designs.

Often, operational requirements are complete only in regard to the active operational functions of the system and do not cover all the constraints and external interactions that the system must comply with during its production, transportation, installation, and operational maintenance. To ensure that these interactions are treated as fully as possible at this stage of development, it is necessary to perform a life cycle analysis and to provide scenarios that represent these interactions.

All requirements must be associated with and traceable to fulfilling the operational objectives of the user. This includes understanding who will be using the system and how it will be operated. Compliance with this guideline helps to minimize unnecessary or extraneous requirements. It also serves as a good communication link between the customer and developer when particular requirements subsequently lead to complex design problems or difficult technical trade-offs.

The essential needs of the customer must be given top priority. If the needs analysis phase has been done correctly, requirements stemming from these needs will be clearly understood by all concerned. When design conflicts occur later in development, a review of these primary objectives can often provide useful guidance for making a decision.

Beyond the above primary or essential requirements, there are always those capabilities that are desirable if they prove to be readily achievable and affordable. Requirements that are essential should be separately distinguished from those that are desirable but not truly necessary for the success of the primary mission. Often, preferences of the customer come through as hard and fast requirements, when they are meant to be desirable features. Examples of desirable requirements are those that provide an additional performance capability or design margin. There should be some indication of cost and risk associated with each desirable requirement so that an informed prioritization can be made. The discrimination between essential and desirable requirements and their prioritization is a key systems engineering function.

The Triumvirate of Conceptual Design

Above, we mentioned the use of the six primitive interrogatives in developing requirements. We also discussed that operational requirements' focus on "why" and functional requirements' focus on the "what" (along with performance requirements' focus on its associated interrogative, "how much"). So, if the two sets of requirements focus on why and what, then where does the analyst go to understand the other four primitive interrogatives? The answer lies with what we call the triumvirate of conceptual design, illustrated in Figure 7.4.

Three products are needed to describe the six interrogatives that collectively could be considered a system concept. The requirements (all three types we have addressed in detail up to this point) address why and what. A new product, the operational concept, sometimes referred to as a concept of operations (CONOPS), addresses how and who. And a description of the operational context, sometimes referred to as scenarios, addresses where and when. Of course, there is a significant overlap between the three, and often two or more of these products are combined into a single document.

Operational Concept (CONOPS)

Although the two terms are often used synonymously, in truth, an operational concept is a broader description of a capability that encompasses multiple systems. It tends to

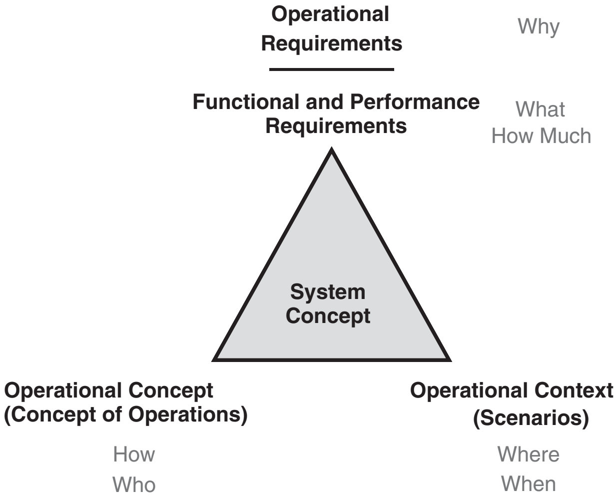

Figure 7.4. Triumvirate of conceptual design.

describe how a large collection of systems will operate. Examples would include an operational concept for the U.S. Transportation System (or even a subsystem of the whole system). In this case, "system" does not refer to a single system but a collection of systems. Another example would be an operational concept for an oil refinery—again referring to how a collection of systems would operate together. When referring to a single system, the term CONOPS is generally used. A further distinction relates to scenarios. An operational concept is sufficiently broad to be scenario independent. A CONOPS tends to relate to a single scenario or a set of related scenarios.

Operational concepts are useful since requirements should avoid prescribing how they should be fulfilled. Requirements documents risk inadvertently barring an especially favorable solution. However, a set of operational requirements alone is often insufficient to constrain the system solutions to the types desired. For example, the operational requirements for defending an airplane against terrorist attack could conceivably be met by counterweapons, passenger surveillance, or sensor technology. In a particular program initiative, the requirements would be constrained by adding a CONOPS, which would describe the general type of counterweapons that are to be considered. This extension of the operational requirement adds constraints, which express the customer's expectation for the anticipated system development.

The term *CONOPS* is quite general. The components of a CONOPS usually include

1. mission descriptions, with success criteria;
2. relationships with other systems or entities;
3. information sources and destinations; and
4. other relationships or constraints.

The CONOPS should be considered as an addition to the operational requirements. It defines the general approach, though not a specific implementation, to the desired system, thereby eliminating undesired approaches. In this way, the CONOPS clarifies the intended goal of the system.

The CONOPS should be prepared by the customer organization or by an agent of the customer and should be available prior to the beginning of the concept definition phase. Thereafter, it should be a "living" document, together with the operational requirements document.

Operational Context Description (Scenarios)

A description of an operational context is the last piece of the triumvirate in defining the system concept. This description (as depicted in Fig. 7.4) focuses on the where and when. Specifically, an operational context description describes the environment within which the system is expected to operate. A specific instantiation of this context is known as a scenario.

A scenario can be defined as "a sequence of events involving detailed plans of one or more participants and a description of the physical, social, economic, military, and political environment in which these events occur." With respect to system development, scenarios are typically projected into the future to provide designers and engineers a context for the system description and design.

Most scenarios include at least five elements:

1. *Mission Objectives:* a description of the overall mission with success criteria. The reader should notice this is the same as one of the components of a CONOPS. The mission can be of any type, for example, military, economic, social, or political.
2. *Friendly Parties:* a description of friendly parties and systems, and the relationships among those parties and systems.
3. *Threat Actions (and Plans):* a description of actions and objectives of threat forces. These threats need not be human; they could be natural (e.g., volcano eruption).
4. *Environment:* a description of the physical environment germane to the mission and system.
5. *Sequence of Events:* a description of individual events along a timeline. These event descriptions should not specify detailed system implementation details.

Scenarios come in all sizes and flavors. The type of scenario is determined by the system in questions and the problem being examined. Figure 7.5 shows different levels of scenarios that might be needed in a system development effort. During the early phases (needs analysis and concept exploration), the scenarios tend to be higher levels, near the top of the pyramid. As the development effort transitions to later phases, more detail is available as the design improves, and lower-level scenarios are used in engineering

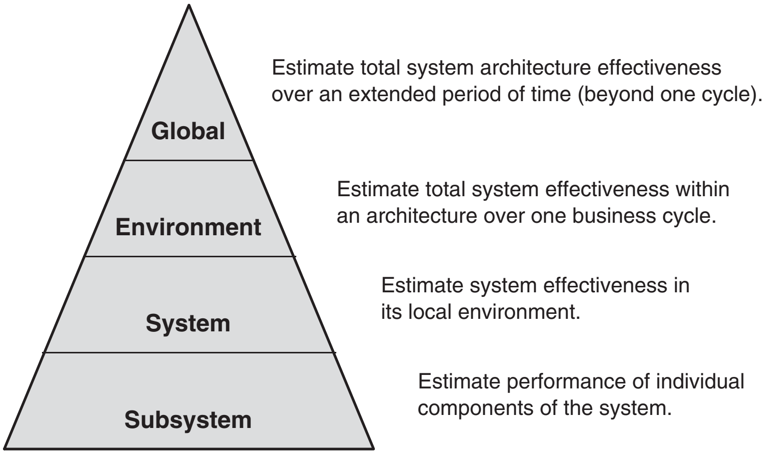

Figure 7.5. Hierarchy of scenarios.

analyses. High-level scenarios continue to be used throughout to estimate the overall system effectiveness as the design matures.

Analysis of Alternatives

The needs analysis phase is usually conducted without the benefit of a well-organized and funded effort. In such cases, the operational requirements that are formulated during this phase are necessarily a preliminary and incomplete definition of the full mission objectives. Therefore, an essential part of the concept exploration phase is to develop the operational requirements into a complete and self-consistent framework as a basis for developing an effective operational system.

For the above reason, before initiating a major program, one or more studies are generally carried out to refine the operational requirements by modeling the interaction of operational scenarios. One of the common designations for such studies is "analysis of alternatives" because they involve the definition of a range of alternative system approaches to the general operational mission, and a comparative evaluation of their operational effectiveness. Such analyses define the realistic limits of expected operational effectiveness for the postulated operational situation and provide the framework for a set of complete, consistent, and realistic operational requirements.

Guidelines for Defining Alternative Concepts. As noted in the next section, conceiving new candidate approaches to satisfying a set of requirements is an inductive process and hence requires a leap of the imagination. For such a process, it is helpful to postulate some guidelines for selecting alternatives:

1. Start with the existing (predecessor) system as a baseline.
2. Partition the system into its major subsystems.

3. Postulate alternatives that replace one or more of the subsystems essential to the mission with an advanced, less costly, or otherwise superior version.
4. Vary the chosen subsystems (or superior version) singly or in combination.
5. Consider modified architectures, if appropriate.
6. Continue until you have a total of four to six meaningful alternatives.

Effectiveness Simulation. Where the analysis of alternatives involves complex systems, the analysis often requires the use of a computer simulation that measures the effectiveness of a model of a system concept in dealing with a model scenario of the system environment. Chapter 9 contains a brief description of the character and application of system effectiveness simulation.

The advantage of computer simulation is that it is possible to provide controls that vary the behavior of a selected system and environmental parameters in order to study their effect on the overall system behavior. This feature is especially valuable in characterizing the effect of operational and performance requirements on the system architecture necessary to satisfy them, and in turn, establishing practical bounds on the requirements. A range of solutions of varying capability and cost can be considered. Every particular application has its own key variables that can be called into play.

7.3 PERFORMANCE REQUIREMENTS FORMULATION

As noted previously, in the course of developing a new system, it is necessary to transform the system operational requirements, which are stated as required outcomes of system action, into a set of system performance requirements, which are stated in terms of engineering characteristics. This step is essential to permit subsequent stages of system development to be based on and evaluated in engineering rather than operational terms. Thus, system functional performance requirements represent the transition from operational to engineering terms of reference.

Derivation of Subsystem Functions

In deriving performance requirements from operational objectives, it is first necessary to identify the major functions that the system must perform to carry out the prescribed operational actions. That means, for example, that if a system is needed to transport passengers to such destinations as they may wish along existing roadways, its functional elements must include, among others, a source of power, a structure to house the passengers, a power-transmitting interface with the roadway, and operator-activated controls of locomotion and direction. Expressed in functional terms (verb–object), these elements might be called “power vehicle,” “house passengers,” “transmit power to roadway,” “control locomotion,” and “control direction.”

As described in Chapter 6, a beginning in this process has already been made in the preceding phase. However, a more definitive process is needed to establish specific performance parameters. Correspondingly, as seen in Table 4.1, during this phase, the functional definition needs to be carried a step further, that is, to a definition of subsystem functions, and to the visualization of the functional and associated physical components, which collectively can provide these subsystem functions.

The Nondeterministic Nature of System Development

The derivation of performance requirements from desired operational outcomes is far from straightforward. This is because, like other steps in the system materialization process, the design approach is inductive rather than deductive, and hence not directly reversible. In going from the more general operational requirements to the more specifically defining system performance requirements, it is necessary to fill in many details that were not explicitly called out in the operational requirements. This can obviously be done in a variety of ways, meaning that more than one system configuration can, in principle, satisfy a given set of system requirements. This is also why in the system development process the selection of the "best" system design at a given level of materialization is accomplished by trade-off analysis, using a predefined set of evaluation criteria.

The above process is exactly the same as that used in inductive reasoning. For example, in designing a new automobile to achieve an operational goal of 600 mi on a tank of gasoline, one could presumably make its engine extremely efficient, or give it a very large gasoline tank, or make the body very light, or some combination of these characteristics. Which combination of these design approaches is selected would depend on the introduction of other factors, such as relative cost, development risk, passenger capacity, safety, and many others.

This process can also be understood by considering a deductive operation, as, for example, performance analysis. Given a specific system design, the system's performance may be deduced unambiguously from the characteristics of its components by first breaking down component functions, then by calculating their individual performance parameters, and finally by aggregating these into measures of the performance of the system as a whole. The reverse of this deductive process is, therefore, inductive and consequently nondeterministic.

One can see from the preceding discussion that, given a set of operational requirements, there is no direct (deductive) method of inferring a corresponding unique set of system performance characteristics that are necessary and sufficient to specify the requirements for a system to satisfy the operational needs. Instead, one must rely on experience-based heuristics, and to a large extent, on a trial and error approach. This is accomplished through a process in which a variety of different system configurations are tentatively defined, their performance characteristics are deduced by analysis or data collection, and these are subjected to effectiveness analysis to establish those characteristics required to meet the operational requirements. The above process is described in greater detail in the next section.

Functional Exploration and Allocation

The exploration of potential system configurations is performed at both the functional and physical levels. The range of different functional approaches that produce behavior suitable to meet the system operational requirements is generally much more limited than the possibilities for different physical implementations. However, there are often several significantly different ways of obtaining the called for operational actions. It is important that the performance characteristics of these different functional approaches be considered in setting the bounds on system performance requirements.

As noted earlier in Figure 7.2, one of the outputs of this step is the allocation of operational functions to individual subsystems. This is important in order to set the stage for the next step, in which the basic physical building block components may be visualized as part of the exploration of implementation concepts. These two steps are very tightly bound through iterative loops, as shown in the figure. Two important inputs to the functional allocation process are the predecessor system and functional building blocks. In most cases, the functions performed by the subsystems of the predecessor system will largely carry over to the new system. Accordingly, the predecessor system is especially useful as a point of departure in defining a functional architecture for the new system. And since each functional building block is associated with both a set of performance characteristics and a particular type of physical component, the building blocks can be used to establish the selection and interconnection of elementary functions and the associated components needed to provide the prescribed subsystem functions.

To aid in the process of identifying those system functions responsible for its operational characteristics, recall from Chapter 3 that functional media can be classed into four basic types: signals, data, material, and energy. The process addresses the following series of questions:

1. Are there operational objectives that require sensing or communications? If so, this means that signal input, processing, and output functions must be involved.
2. Does the system require information to control its operation? If so, how are data generated, processed, stored, or otherwise used?
3. Does system operation involve structures or machinery to house, support, or process materials? If so, what operations contain, support, process, or manipulate material elements?
4. Does the system require energy to activate, move, power, or otherwise provide necessary motion or heat?

Furthermore, functions can be divided again into three categories: input, transformative, and output. Input functions relate to the processes of sensing and inputting signals, data, material, and energy into the system. Output functions relate to the processes of interpreting, displaying, synthesizing, and outputting signals, data, material, and energy out of the system. Transformative functions relate to the processes of transforming the inputs to the outputs of the four types of functional media. Of course, for complex systems, the number of transformative functions may be quite large, and has successive "sequences" of transformations. Figure 7.6 depicts the concept of this two-dimensional construct, function category versus functional media.

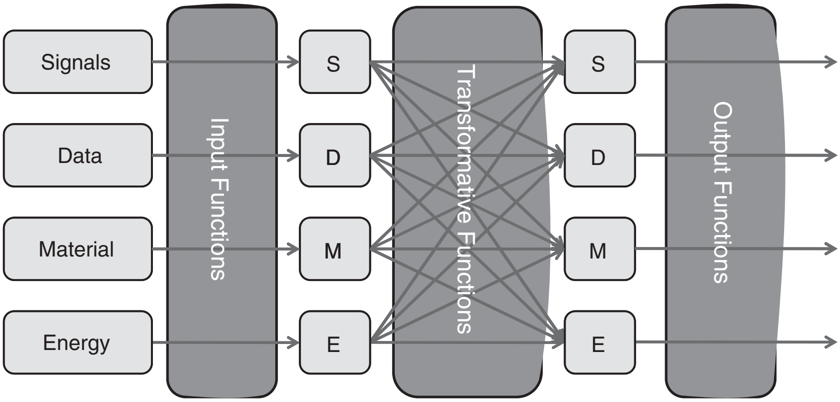

Figure 7.6. Function category versus functional media.

In constructing an initial function list, it helps to identify inputs and outputs (as described in Chapter 3). This list directly leads the engineer to a list of input and output functions. The transformative functions may be easier to identify when examining them in the light of a system's inputs and outputs.

As an example, while acknowledging it is not a complex system, consider a common coffeemaker (without any frills). By observation, an analyst can identify the necessary inputs:

- *Signals:* user commands (which we will simply identify as "on" and "off")
- *Data:* none
- *Materials:* fresh coffee grinds, filter, and water
- *Energy:* electricity
- *Forces:* mechanical support

Outputs can also be easily identified:

- *Signals:* status (which we will simply identify as on and off)
- *Data:* none
- *Materials:* brewed coffee, used filter, used coffee grinds
- *Energy:* heat
- *Forces:* none

Identifying inputs and outputs assists the analyst in identifying functions. Input functions will directly proceed from the input list (deductive reasoning). Output functions will directly proceed from the output list (deductive as well). The transformative functions will be more difficult to identify since doing so relies on inductive reasoning.

However, we now have a guide to this inductive process: we know that we must transform the six inputs into the five outputs.

This line of inquiry normally reveals all operationally significant functions and permits them to be grouped in relation to specific operational objectives. Further, this grouping naturally tends to bring together the elements of different subsystems, which are the first-level building blocks of the system itself. The above strategy is also appropriate even if the basic configuration is derived from a predecessor system because its generic and systematic approach tends to reveal elements that might otherwise be overlooked. In the coffeemaker case, we can focus on transforming the input materials and signals into output materials and signals. In other words, we can identify functions by answering the question "How do we transform fresh coffee grinds, a filter, water, and an on/off command into brewed coffee, a used filter, used coffee grinds, and a status?"

Keeping the list of functions minimal and high level, and using the verb–object syntax, an example list pertaining to the coffeemaker could be

Input Functions

1. Accept user command (on/off)
2. Receive coffee materials
3. Distribute electricity
4. Distribute weight

Transformative Functions

5. Heat water
6. Mix hot water with coffee grinds
7. Filter out coffee grinds
8. Warm brewed coffee

Output Functions

9. Provide status
10. Facilitate removal of materials
11. Dissipate heat

Can you map the inputs and outputs to one or more functions? Can you identify how the inputs are transformed into the outputs? Since a coffeemaker is a very simple system, the number of transformative functions was low. But keep in mind that regardless of a system's complexity, a top-level function list with about 5–12 functions can always be identified. So a complex system may have a large hierarchy of functions, but any system can be aggregated into an appropriate set of top-level functions.

Formulation of Performance Characteristics. As noted above, the objective of the concept exploration phase is to derive a set of system performance characteristics that are both necessary and sufficient. This means that a system possessing them will satisfy the following criteria:

1. A system that meets the system operational requirements and is technically feasible and affordable will comply with the performance characteristics.
2. A system that possesses these characteristics will meet the system operational requirements and can be designed to be technically feasible and affordable.

The condition that the set of performance requirements must be necessary as well as sufficient is essential to ensure that they do not inadvertently exclude a system concept that may be especially advantageous compared to others just because it may take an unusual approach to a particular system function. This often happens when the performance requirements are derived in part from a predecessor system and carry over features that are not essential to its operational behavior. It also happens when there is a preconceived notion of how a particular operational action should be translated into a system function.

For the above reasons, the definition of performance characteristics needs to be an exploratory and iterative process, as shown in Figure 7.2. In particular, if there are alternative functional approaches to an operational action, they should all be reflected in the performance characteristics, at least until some may be eliminated in the implementation and validation steps in the process.

Incompatible Operational Requirements. It should be noted that a given set of operational requirements does not always lead to feasible performance characteristics. In Chapter 6, the automobile was mentioned as a system that was required to undergo significant changes because of government-imposed regulations concerning safety, fuel economy, and pollution control. Initially, these areas of regulation were independently developed. Each set of requirements was imposed solely on the basis of a particular need, with little regard for either the associated engineering problems or other competing needs. When these regulations were subjected to engineering analysis, it was shown that they were not collectively feasible within the practical technology available at that time. Also, the investment in development and production would result in a per-unit cost far in excess of the then current automobile prices. The basic reason for these problems was that the available pollution controls necessary to meet emission requirements resulted in lowered fuel economy, while the weight reduction necessary to meet the required fuel economy defeated the safety requirements. In other words, the three independent sets of operational requirements turned out to be incompatible because no one had initially considered their combined impact on the design. Note that in this instance, an analysis of the requirements did not depend on a detailed design study since simply examining the design concepts readily revealed the conflicts.

Example: Concepts for a New Aircraft. An instructive example of concept exploration is illustrated by the acquisition of a new commercial aircraft. Assume for this discussion that an airline company serves short to medium domestic routes using two-engine propeller-driven aircraft. Many of the airports it serves have relatively short runways. This arrangement has worked well for a number of years. The problem that has become more and more apparent is that because of increasing maintenance and fuel expenses, the cost per passenger mile has increased to the point where the business is marginally profitable. The company is therefore considering a major change in its aircraft. In essence, the airline's need is to lower the cost per passenger mile to some acceptable value and to maintain its competitive edge in short-route service.

The company approaches several aircraft manufacturers for a preliminary discussion of a new or modified airplane to meet its needs. The discussions indicate that there are several options available. Three such options are the following:

1. A stretched aircraft body and increased power. Engines of the appropriate form and fit exist for such a configuration. This option permits a quick, relatively low-cost upgrade, which increases the number of passengers per aircraft, thereby lowering the overall cost per passenger mile.
2. A new, larger, four-engine propeller aircraft, using state-of-the-art technology. This option offers a good profit return in the near term. It is reasonably low risk, but the total useful life of the aircraft is not well-known, and growth potential is limited.
3. A jet-powered aircraft that is capable of takeoff and landing at most, but not necessarily all, of the current airports being served. This option permits a significant increase in passengers per airplane and opens up the possibility of competing for new, longer routes. This is also the most expensive option. Because of the inherent lower maintenance and fuel costs of jet engines relative to propeller engines, operating costs for this aircraft are attractive, but some existing routes will be lost.

It is evident that the final choice will require considerable expertise and should be based on a competition among interested manufacturers. The airline engages the services of an engineering consulting company to help its staff prepare a set of aircraft performance requirements that can serve as a basis for competitive bids and to assist in the selection process.

In exploring the above and related options, the alternative functional approaches are considered first. These appear to center on the choice between staying with propeller engines, an option that retains the basic features of the present aircraft, or moving to jet engines, which offer considerable operating economies. However, the latter is a major departure from the current system and will also affect its operational capabilities. To permit this choice to be left open to the bidders, the performance requirements such as runway length, cruising speed, and cruising altitude will need to be sufficiently broad to accommodate these two quite different functional approaches.

Requirements Formulation by Integrated Product Teams (IPTs)

As noted earlier, the responsibility for defining the performance requirements of a new product is that of the customer, or in the case of government programs, that of the acquisition agency. However, the organization of the process and its primary participants varies greatly with the nature of the product, the magnitude of the development, and the customer auspices.

As in the case of all acquisition practices, the Department of Defense (DoD) has had the most experience with various methods for organizing the acquisition process. A recent practice introduced by DoD is the use of IPTs throughout the acquisition process. IPTs are intended to bring a number of benefits to the process:

1. They bring senior industry participants into the system conceptual design process at the earliest opportunity, thereby educating them in the operational needs and injecting their ideas during the formative stages of the development.
2. They bring together the different disciplines and specialty engineering viewpoints throughout the development.
3. They capitalize on the motivational advantages of team collaboration and consensus building.
4. They bring advanced technology and COTS knowledge to bear on system design approaches.

As in the case of any organization, the success of this approach is highly dependent on the experience and interpersonal skills of the participants, as well as on the leadership qualities of the persons responsible for team organization. And perhaps even more important is the systems engineering experience of the team leaders and members. Without this, the majority of the team members, who tend to be specialists, will not be able to communicate effectively and hence the IPT will not achieve its objectives.

7.4 IMPLEMENTATION OF CONCEPT EXPLORATION

The previous section discussed the exploration of alternative functional approaches—concepts in which the nature of the activities involved differs from one case to the next. The physical implementation of such concepts involves the examination of different technological approaches, generally offering a more diverse source of alternatives. As in the case of examining alternative functional concepts, the objective of exploring implementation concepts is to consider a sufficient variety of approaches to support the definition of a set of system performance requirements that are feasible of realization in practice and do not inadvertently preclude the application of an otherwise desirable concept. To that end, the exploration of system concepts needs to be broadly based.

Alternative Implementation Concepts

The predecessor system, where one exists, forms one end of the spectrum to be explored. Given the operational deficiencies of the predecessor system to meet projected needs, modifications to the current system concept should first be explored with a view to eliminating these deficiencies. Such concepts have the advantage of being relatively easier to assess from the standpoint of performance, development risk, and cost than are radically different approaches. They can also generally be implemented faster, more cheaply, and with less risk than innovative concepts. On the other hand, they are likely to have severely limited growth potential.

The other end of the spectrum is represented by innovative technical approaches featuring advanced technology. For example, the application of powerful, modern microprocessors might permit extensive automation of presently employed manual operations. These concepts are generally riskier and more expensive to implement but offer large incremental improvements or cost reduction and greater growth potential. In between are intermediate or hybrid concepts, including those defined in the needs analysis phase for demonstrating the feasibility of meeting the proposed system needs.

Many techniques exist for developing new and innovative concepts. Perhaps the oldest is brainstorming, individually and within a group. Within the concept of brainstorming, several modern methods, or variations, to the old fashioned, largely unstructured brainstorming process have risen. One of our favorite techniques, which engineers may not be familiar with (but nonengineering practitioners may be), is Mind Maps. This particular technique uses visual images to assist in the brainstorming of new ideas. A simple Web search will point the reader to multiple Web sites describing the technique.

The natural temptation to focus quickly on a single concept or "point design" approach can easily preclude the identification of other potentially advantageous approaches based on fundamentally different concepts. Accordingly, several concepts spanning a range of possible design approaches should be defined and investigated. At this stage, it is important to encourage creative thinking. It is permissible, even sometimes desirable, to include some concepts that do not meet all of the requirements; otherwise, a superior alternative may be passed by because it fails to meet what may turn out to be a relatively arbitrary requirement. Just as in the needs analysis phase, negotiations with the customer regarding which requirements are really necessary and which are not can often make a significant difference in cost and risk factors while having minimal impact on performance.

Example: Concept Exploration for a New Aircraft. Returning to the example introduced in the previous section, it will be recalled that two principal functional options were explored to meet the need of the airline company: a propeller-driven and a jet-driven aircraft. It remains to explore alternative physical implementations of each of these options. As is usually the case, these are more numerous than the basic functional alternatives.

In the period since the airline's present fleet was acquired, a host of technological advances have occurred. For example, automation has become more widespread, especially in autopilots and navigation systems. Changes in safety requirements, such as for deicing provisions, must also be examined to identify those performance characteristics that should be called out. In exploring alternative implementations, the main features of each candidate system must first be analyzed to see if they are conceptually achievable. At this stage of development, a detailed design analysis is usually not possible because the concept is not yet sufficiently formulated. However, based on previous experience and engineering judgment, someone, usually the systems engineer, must decide whether or not the concept as proposed is likely to be achievable within the given bounds of time, cost, and risk.

There are numerous other options and variations of the above examples. It is noted that all the cited options have pros and cons, which typically leave the customer with no obvious choice. Note also that the option to use jet aircraft may partially violate the operational requirement that short-route capability be maintained. However, as noted earlier, it is not at all unusual at this stage to consider options that do not meet all the initial requirements to ensure that no desirable option is overlooked. The airline may decide that the loss of some routes is more than compensated for by the advantages to the overall system of using jet aircraft.

It is also important to note that the entire system life cycle must be considered in exploring alternatives. For example, while the jet option offers a number of performance advantages, it will require a substantial investment in training and logistic support facilities. Thus, assessment of these supporting functions must be included in formulating system requirements. In order to be a "smart buyer," the airline needs to have a staff well versed in aircraft characteristics, as well as in the business of running an airline, and access to consultants or engineering services organizations capable of carrying out the analyses involved in developing the requisite set of performance requirements.

Preferred System. Although in most cases it is best to refrain from picking a superior system concept prematurely, there are instances where it is permissible for the requirements definition effort to identify a so-called preferred system, in addition to considering a number of other viable system alternatives. Preference for a system or subsystem may be set forth when significant advanced development work has taken place and has produced very promising results in anticipation of future upgrades to the current system. Such work is often conducted or sponsored by the customer. Another justifying factor may be when there has been a recent major technological breakthrough, which promises high gains in performance at an acceptable risk. The idea of a preferred system approach is that subsystem analysis can start building on this concept, thereby saving time and cost. Of course, further analysis may show the favored approach not to be as desirable as predicted.

Technology Development

Whether the origin of a new system is needs driven or technology driven, the great majority of new systems have been brought into being, directly or indirectly, as a result of technological growth. In the process of exploring potential concepts for the satisfaction of a newly established need, a primary input is derived from what is called the technology base, which means the sum total of the then existing technology. It is, therefore, important for systems engineers to understand the nature and sources of technological advances that may be pertinent to a proposed system development.

System-oriented exploratory R & D can be distinguished according to whether it relates to new needs-driven or technology-driven systems. The former is mainly directed to gaining a firm understanding of the operational environment and the factors underlying the increased need for the new system.

The latter is usually focused on extending and quantifying the knowledge base for the new technology and its application to the new system objectives. In both instances, the objective is to generate a firm technical base for the projected system development, thus clarifying the criteria for selecting specific implementation concepts and transforming unknown characteristics and relationships into knowns.

Both industry and government support numerous programs of R & D on components, devices, materials, and fabrication techniques, which offer significant gains in performance or cost. For instance, most large automobile manufacturers have ongoing programs to develop more efficient engines, electrically powered vehicles, automated fuel controls, lighter and stronger bodies, and a host of other improvements that are calculated to enhance their future competitive position. In recent years, the greatest amount of technology growth has been in the electronics industry, especially computers and communication equipment, which in turn has driven the explosive growth of information systems and automation generally.

In government-sponsored R & D, there is also a continuing large-scale effort, mainly among government contractors, laboratories, and universities, directed toward the development of technologies of direct interest to the government. These cover many diverse applications, and their scope is almost as broad as that of commercial R & D. As has been noted previously, defense contractors are permitted to charge a percentage of their revenues from government contracts to IRAD as allowable overhead. A large fraction of such funds is devoted to activities that relate to potential new system developments. In addition, there is a specific category in the Congressional Research, Development, Test and Evaluation (RDT&E) appropriation, designated Research and Exploratory Development, which funds specific R & D proposals to the military services. Such projects are not intended to directly support specific new system developments but do have to be justified as contributing to existing mission areas.

Performance Characteristics

The derivation of performance characteristics by the exploration of implementation concepts can be thought of as consisting of a combination of two analytical processes: performance analysis and effectiveness analysis. Performance analysis derives a set of performance parameters that characterize each candidate concept. Effectiveness analysis determines whether or not a candidate concept meets the operational requirements and, if not, how the concept needs to be changed to do so. It employs an effectiveness model that is used to evaluate the performance of a conceptual system design in terms of a selected set of criteria or measures of effectiveness. This is a similar model to that used in the previous phase and to the one employed in the next step, the validation of performance requirements. The main difference in its use in the above applications is the level of detail and rigor.

Performance Analysis. The performance analysis part of the process is used to derive a set of relevant performance characteristics for each candidate system concept that has been found to satisfy the effectiveness criteria. The issue of relevancy arises because a full description of any complex system will involve many parameters, some of which may not be directly related to its primary mission. For example, some features, such as the ability of an aircraft search radar device to track some particular coded beacon transponder, might be included only to facilitate system test or calibration. Therefore, the performance analysis process must extract from the identified system characteristics only those that directly affect the system's operational effectiveness. At the same time, care must be taken to include all characteristics that can impact effectiveness under one or another particular operating condition.

The problem of irrelevant characteristics is especially likely to occur when the concept for a particular subsystem has been derived from the design of an existing subsystem employed in a different application. For example, a relatively high value of the maximum rate of train or elevation for a radar antenna assembly might not be relevant to the application now being examined. Thus, the derived model should not reflect this requirement unless it is a determining factor in the overall subsystem design concept. In short, as stated previously, the defined set of characteristics must be both necessary and sufficient to facilitate a valid determination of effectiveness for each candidate system concept.

Constraints. At this phase of the project, the emphasis will naturally be focused on active system performance characteristics and functions to achieve them. However, it is essential that other relevant performance characteristics not be overlooked, especially the interfaces and interactions with other systems or parts of systems, which will invariably place constraints on the new system. These constraints may affect physical form and fit, weight and power, schedules (e.g., a launch date), mandated software tools, operating frequencies, operator training, and so on. While constraints of this type will be dealt with in great detail later in the development process, it is not too soon to recognize their impact during the process of requirements definition. The immediate benefit of early attention to such problems is that conflicting concepts can be filtered out, leaving more time for analysis of the more promising approaches.

To accomplish the above objectives, it is necessary to consider the complete system life cycle. To a large extent, the constraints on the system will not depend on the specific system architecture. For example, environmental conditions of temperature, humidity, shock vibration, and so forth, for a great part of the system life cycle are often the same for any candidate system concept. Omission of any constraints such as these may result in serious deficiencies in the system design, which would adversely impact performance and operability.

7.5 PERFORMANCE REQUIREMENTS VALIDATION

Having derived the operationally significant performance characteristics for several feasible alternative concepts, all of which appear to be capable of meeting the system operational requirements, the next step is to refine and integrate them into a singular set to serve as a basis for the preparation of formal system performance requirements.

As stated earlier, these performance requirements, stated in engineering units, provide an unambiguous basis for the ensuing phases of system development, up to the stage where the actual system can be tested in a realistic environment.

The operations involved in the refinement and validation of system performance requirements can be thought of as two tightly coupled processes—an integration process, which compares and combines the performance characteristics of the feasible alternative concepts, and an effectiveness analysis process, which evaluates the validity of the integrated characteristics in terms of the operational requirements.

Performance Characteristics Integration

The integration process serves to select and refine those characteristics of the different system concepts examined in the exploration process that are necessary and sufficient to define a system that will possess the essential operational characteristics. Regardless of the analytical tools that may be available, this process requires the highest level of systems engineering judgment.

This and other processes in this phase can benefit greatly by the participation of systems engineers with experience with the predecessor system, which has been mentioned a number of times previously. The knowledge and database that comes with that system is an invaluable source of information for developing new requirements and concepts. In many cases, some of the key engineers and managers who directed its development may still be available to contribute to the development of new requirements and concepts. They may not only be aware of the current deficiencies but are likely to have considered various improvements. Additionally, they are probably aware of what the customer really wants, based on their knowledge of operational factors over a number of years. Just one key systems engineer with this background can provide significant help. Experienced people of this type will also have an educated "gut feel" about the viability of the requirements and concepts that are being considered. Their help, at least as consultants, will not alleviate the need for requirements analysis, but it may quickly point the effort in the right direction and avoid blind alleys that might otherwise be pursued.

Performance Characteristics Validation

The final steps in the process are to validate the derived performance characteristics against the operational requirements and constraints and to convert them into the form of a requirements document. Ideally, the performance characteristics derived from the refinement step will have been obtained from concepts validated in the implementation concept exploration process. However, it is likely that the effort to remove irrelevant or redundant characteristics in the integration step, and to add external constraints not present in the effectiveness model, will have significantly altered the resultant set of characteristics. Hence, it is essential to subject them once more to an effectiveness analysis to verify their compliance with the operational requirements. The effectiveness model in the above step should generally be more rigorous and detailed than models used in previous steps so as to ensure that the final product does not contain deficiencies due to omission of important evaluation criteria.

The above processes operate in closed-loop fashion until a self-consistent set of *system performance characteristics* that meets the following objectives is obtained:

1. They define what the system must do, and how well, but not how the system should do it.
2. They define characteristics in engineering terms that can be verified by analytical means or experimental tests, so as to constitute a basis for ensuing engineering phases of system development.
3. They completely and accurately reflect the system operational requirements and constraints, including external interfaces and interactions, so that if a system possesses the stated characteristics, it will satisfy the operational requirements.

Requirements Documentation

To convert the system performance characteristics into a requirements document involves skillful organization and editing. Since the system performance requirements will be used as the primary basis for the ensuing concept definition phase and its successors, it is most important that this document be clear, consistent, and complete. However, it is equally important to recognize that it is not carved in stone but is a living document, which will continue to evolve and improve as the system is developed and tested.

In a need-driven system development in which it is intended to compete the concept definition phase among a number of bidders, the system performance requirements are a primary component of the competitive solicitation, along with a complete statement of all other conditions and constraints. Such a solicitation is often circulated in draft form among potential bidders to help ensure its completeness and clarity.

In a technology-driven system development in which the same commercial company that will carry out the definition and subsequent phases conducts the exploratory phase, the end product typically serves as a basis for deciding whether or not to authorize and fund a concept definition phase preliminary to engineering development. For this purpose, the requirements document typically includes a thorough description of the most attractive alternative concepts investigated, evidence of their feasibility, market studies validating the need for a new system, and estimates of development, production, and market introduction costs.

7.6 SUMMARY

Developing the System Requirements

The objectives of the concept exploration phase (as defined here) are to explore alternative concepts to derive common characteristics and to convert the operationally oriented system view into an engineering-oriented view. Outputs of concept exploration are (1) system performance requirements, (2) a system architecture down to the subsystem level, and (3) alternative system concepts.

Activities that comprise concept exploration are the following:

- *Operational Requirements Analysis*—ensuring completeness and consistency;
- *Implementation Concept Exploration*—refining functional characteristics;
- *Performance Requirements Formulation*—deriving functions and parameters; and
- *Performance Requirements Validation*—ensuring operational validity.

Operational Requirements Analysis

Requirements development involves four basic steps: elicitation, analysis, validation, and documentation. These steps will, done correctly, lead to a robust set of well-articulated requirements.

Generating operational-level requirements usually involves analyses of alternative concepts, typically involving effectiveness models and simulations. In order to conduct these important analyses, three components are necessary: an initial set of operational requirements, an operational concept for the system in question, and the operational context—a set of operational scenarios depicting the environment.

Performance Requirements Formulation

System development is a nondeterministic process in that it requires an iterative inductive reasoning process, and many possible solutions can satisfy a set of operational requirements. The predecessor system can be of great assistance as it will help define the system functional architecture and the performance of functional building blocks.

Implementation Concept Exploration

Exploration of alternative implementation concepts should

- avoid the "point design syndrome";
- address a broad spectrum of alternatives;
- consider the adaptation of a predecessor system technology;
- consider innovative approaches using advanced technology; and
- assess the performance, risk, cost, and growth potential of each alternative.

Technology development is also an important component of system development. Industry and government support major R & D programs that lead to new technologies. This foundation of technology is typically referred to as the "technical base" and is the source of many innovative concepts.

System performance requirements are developed through analyses to establish the performance parameters of each concept. These requirements are then assessed for conformance with operational requirements and constraints. Sources of these constraints include (1) system operator, maintenance, and test considerations; (2) requirements for interfacing with other systems; (3) externally determined operational environments; and (4) fabrication, transportation, and storage environments.

When completed, system performance requirements define what the system should do, but not how it should do it. They present system characteristics in engineering terms—a necessary and sufficient set reflecting operational requirements and constraints.

Performance Requirements Validation

Performance requirements validation involves two interrelated activities: (1) integration of requirements derived from alternative system concepts and (2) effectiveness analyses to demonstrate satisfaction of the operational requirements. Performance requirements are defined in a living document; requirements are reviewed and updated throughout the system life cycle.

PROBLEMS

7.1 Explain why it is necessary to examine a number of alternative system concepts prior to defining a set of system performance requirements for the purpose of competitive system acquisition. What are the likely results of failing to examine a sufficient range of such concepts?

7.2 To meet future pollution standards, several automobile manufacturers are developing cars powered by electricity. Which major components of gasoline-powered automobiles would you expect to be retained with minor changes? Which ones would probably be substantially changed? Which would be new? (Do not consider components not directly associated with the automobile's primary functions, such as entertainment, automatic cruise control, power seats and windows, and air bags.)

7.3 List the characteristics of a set of well-stated operational requirements, that is, the qualities that you would look for in analyzing their adequacy. For each, state what could be the result if a requirement did not have these characteristics.

7.4 In the section of performance requirements formulation, the process of system development is stated to be "nondeterministic." Explain in your own words what is meant by this term. Describe an example of another common process that is nondeterministic.

7.5 Derive the principal functions of a DVD player by following the checklist shown in the subsection Functional Exploration and Allocation. How does each function relate to the operational requirements of the DVD player?

7.6 IPTs are stated to have four main benefits. What specific activities would you expect systems engineers to perform in realizing each of these benefits?

7.7 What role does exploratory R & D conducted prior to the establishment of a formal system acquisition program play in advancing the objective of a system acquisition program? What are the main differences between the organization and funding of R & D programs and system development programs?

7.8 In considering potential system concepts to meet the operational requirements for a new system, there is frequently a particular concept that appears to be an obvious solution to the system requirements. Knowing that premature focusing on a "point solution" is a poor systems engineering practice, describe two approaches for identifying a range of alternative system concepts for consideration.

7.9 (a) Develop a set of operational requirements for a simple lawn tractor. Limit yourself to no more than 15 operational requirements.

(b) Develop a set of performance requirements for the same lawn tractor. Limit yourself to no more than 30 performance requirements.

(c) Based on your experience, write a short paper defining the process of transforming operational requirements to performance requirements.

(d) How would you go about validating the requirements in (b)?

FURTHER READING

B. Blanchard and W. Fabrycky. *System Engineering and Analysis*, Fourth Edition. Prentice Hall, 2006, Chapter 3.

W. P. Chase. *Management of Systems Engineering*. John Wiley, 1974, Chapters 4 and 5.

H. Eisner. *Essentials of Project and Systems Engineering Management*. Wiley, 1997, Chapter 8.

D. K. Hitchins. *Systems Engineering: A 21st Century Systems Methodology*. John Wiley & Sons, 2007, Chapters 5 and 8.

International Council on Systems Engineering. *Systems Engineering Handbook. A Guide for System Life Cycle Processes and Activities*. Version 3.2, July 2010.

K. Kendall and J. Kendall. *Systems Analysis and Design*, Sixth Edition. Prentice Hall, 2003, Chapters 4 and 5.

A. M. Law. *Simulation, Modeling & Analysis*, Fourth Edition. McGraw-Hill, 2007, Chapters 1, 2, and 5.

H. Lykins, S. Friedenthal, and A. Meilich. Adapting UML for an object-oriented systems engineering method (OOSEM). *Proceedings of 10th International Symposium INCOSE*, July 2000.

A. Meilich and M. Rickels. An application of object-oriented systems engineering to an army command and control system: A new approach to integration of systems and software requirements and design. *Proceedings of Ninth International Symposium INCOSE*, June 1999.