2

THE SYSTEM ENGINEERING PROCESS

The *system engineering process* is inherent within the overall system life cycle, as illustrated in Figure 1.12. The emphasis is on a top-down, integrated, life-cycle approach to system design and development, conveyed through the activities identified in blocks 0.1 through 4.6. This approach includes an initial definition of the problem and the identification of a consumer need, the conductance of a feasibility analysis, the development of operational requirements and the maintenance and support concept, accomplishment of a functional analysis, allocation of requirements, and so on. Subsequently, there is the iterative process of assessment and validation and the incorporation of changes for product/process improvement as required. Although the process is more directed to the early stages of system design and development, consideration of the activities in the latter phases of construction/production, operational utilization, and system maintenance and support is essential for understanding the consequences of earlier decisions and establishing benchmarks for the future. In other words, the feedback loop is critical and an integral part of the system engineering process.

This chapter addresses the system engineering process and the basic activities reflected in Figure 2.1. These activities represent a *process* that should be applied each time there is a newly identified requirement for a system. For example, a new requirement may evolve when a new performance factor has been identified; for example, when the production rate in a factory is doubled, the capacity of a transportation vehicle is increased, a radar range is increased, the speed of data transmission is increased, the weight of a product is reduced, and so on. This is not to imply an additional amount of work or excessive costs, contrary to the perception of many that implementation of system engineering requirements is time-consuming and costly. However, it does require a change in thinking, a shift in emphasis in approaching a system design objective, and a new way of doing business.

The steps shown in Figure 2.1 must (of course) be tailored to the system and program requirements. There are also many iterations that occur within. Analyses and

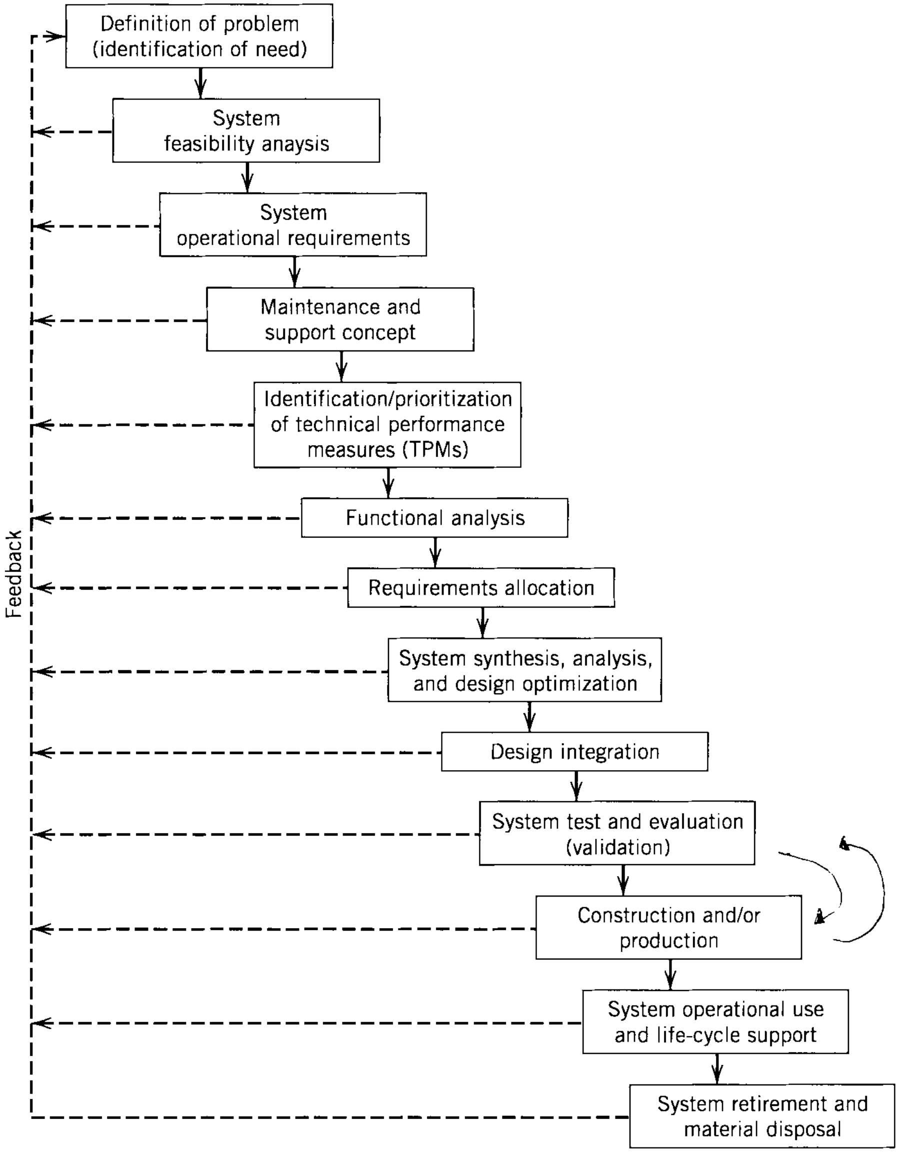


Figure 2.1 The system engineering process in the life cycle.

trade-off studies are accomplished at each stage, functions are identified in several of the blocks, and so on, and it is impossible to show graphically everything that occurs throughout. However, for the purposes of discussion and for better understanding, the steps illustrated in Figure 2.1 are presented herein, in the order indicated.[1]

2.1 DEFINITION OF THE PROBLEM (CURRENT DEFICIENCY)

The system engineering process generally commences with the identification of a "want" or "desire" for something and is based on a real (or perceived) deficiency. For instance, suppose that a current capability is not adequate in terms of meeting certain required performance goals, is not reliable or available when needed, cannot be properly supported, or is too costly to operate. As a result, a new system *requirement* is defined, along with a priority for introduction, the date when the new system capability is required for consumer use, and an estimate of the resources necessary for acquiring the new system capability. To ensure a good start, a complete *description of the problem* should be presented in specific qualitative and quantitative terms, in enough detail to justify progressing to the next step. More specifically, one should pose the following question: What is the nature and magnitude of the problem, and what are the associated risks if the problem is not addressed?

The requirement for identifying the need (as a starting point) may seem to be rather basic or self-evident. However, it often happens that a design effort is initiated as a result of a personal interest or a political whim, without the requirements being adequately defined. In the software area (in particular), there is a tendency to accomplish a lot of coding before identifying the functional need for such. In addition, there are instances in which the engineer sincerely believes that he or she knows what the customer needs, without first having involved the customer in the process. In essence, the attitude design-it-now-fix-it-later often prevails. As a result, it is not uncommon for someone to proceed with the design and ultimately produce a product that really was not wanted (or needed) in the first place. This approach, of course, can be rather expensive.

Defining the problem is sometimes the most difficult part of the process, particularly if one is in a rush to "get going." Yet the number of false starts and the ultimate risks can be rather significant unless a good foundation is laid from the beginning. A complete description of the need, expressed in quantitatively stated *performance* parameters where possible, is essential. It is important that the results reflect a *true* customer requirement, especially in today's environment where resources are limited.

[1]As an example of an "iteration," the *functional analysis* actually commences with the identification of the problem and a description of the functions that are not currently being accomplished; these functions are then established as a basis for the evaluation of possible alternative technology applications in a feasibility analysis; *operational* functions are identified in defining the operational requirements for the system; *maintenance* functions are described in the development of the maintenance concept; and an overall *functional baseline* for the system ultimately evolves from these earlier activities. In other words, the functional analysis is actually accomplished within the first six blocks shown in Figure 2.1, with the appropriate "feedback" as one progresses downward.

2.2 DEVELOPMENT OF CONSUMER NEED

Given the problem definition, a *needs analysis* must be accomplished with the objective of translating a broadly defined "want" into a more specific system-level requirement. At this point, the questions are, What is required of the system in "functional" terms and what specific functions must the system accomplish? Why must they be accomplished? What are the primary functions? What are the secondary functions? What must be accomplished to alleviate the stated deficiency? When must this be accomplished? Where is this to be accomplished and for how long? How many times must this be accomplished? There are many basic questions of this nature, and it is important to describe the consumer (customer) requirements in a *functional* manner in order to avoid a premature commitment to a specific design concept or configuration, and thus the unnecessary expenditure of valuable resources. The ultimate objective is to define the "WHATs" and not the "HOWs."

Accomplishing a needs analysis in a satisfactory manner can best be realized through a team approach involving the customer, the ultimate consumer/user (if different from the customer), the contractor or producer, and major suppliers as appropriate. The objective is to ensure proper communications between the parties involved. The voice of the customer must be heard, and the system developer must respond accordingly. Methods such as conducting surveys, interviews, the use of checklists, the application of such tools as *quality function deployment* (QFD), and related techniques may be employed. As the definition of need is sometimes not apparent in the beginning, there may be several iterations of meetings, interviews, question exchanges, and so on, until there is full agreement.[2]

2.3 SYSTEM FEASIBILITY ANALYSIS

Through a needs analysis, the functions that the system must perform are identified. There may be a single function such as "transport product XYZ from point A to point B," or "communicate between points D, E, and F," or "produce X quantity of Y products by time Z." On the other hand, there may be a number of different functions to be performed, some primary and some secondary. To ensure a good design, all possible functions must be identified, the most rigorous functions being selected as the basis for defining system-level design requirements. It is important that *all* possibilities be addressed to ensure that the proper technologies and components are selected for design consideration.

A *feasibility analysis* is accomplished with the objective of evaluating the different technological approaches that may be considered in responding to the specified

[2]An excellent technique, often utilized as an aid in defining requirements and ensuring that the proper communications exist between the customer/consumer and producer, is the *Quality Function Deployment* (QFD) method. The QFD method was developed initially at the Kobe shipyard of Mitsubishi Heavy Industries, Japan, and has evolved considerably since. It is used to facilitate the translation of a prioritized set of subjective customer requirements into a relevant set of system-level requirements during the conceptual design phase. The application of the QFD method is demonstrated further in Section 2.6.

functional requirements. In considering different design approaches, alternative technology applications are investigated. For instance, in the design of a communications system, should one use fiber-optics technology, cellular, or the conventional hard-wired approach? In designing an aircraft, to what extent should one incorporate composite materials? When designing an automobile, should one apply very high-speed integrated electronic circuitry in certain control applications, or should one select a more conventional electromechanical approach?

It is necessary to (1) identify the various possible design approaches that can be pursued to meet the requirements, (2) evaluate the most likely candidates in terms of performance, effectiveness, logistics requirements, and life-cycle economic criteria, and (3) recommend a preferred approach. The objective is to select an overall *technical* approach, and not to select specific components. There may be many alternatives; however, the number of possibilities must be narrowed down to a few feasible options, consistent with the availability of resources (i.e., manpower, materials, and money).

It is at this early stage in the life cycle (i.e., the conceptual design phase) where major decisions are made relative to adopting a specific design approach. When there is not enough information available, a research activity may be initiated with the objective of developing new methods/techniques for specific applications. In some programs, the completion of applied research tasks and preliminary design activity is accomplished sequentially, whereas in other situations, there may be a number of different mini-projects under way at the same time.

The results of the feasibility analysis will have a significant impact not only on the operational characteristics of a system, but on the production and maintenance support requirements as well. The selection (and application) of a given technology has reliability and maintainability implications, may significantly affect the requirements for spare parts and test equipment, may impact manufacturing methods, and will certainly impact life-cycle cost.

With the early feasibility analysis being so critical and having such a large impact on the follow-on system design and development activity, the role of the system engineer becomes important. In most situations, the detailed investigations and evaluation efforts leading to specific design approaches are highly technical and are accomplished by specialists in a given engineering discipline. Often, these specialists are not oriented to the "system" as an overall entity, or its manufacturing process, or its maintenance and support capability, or the factors affecting life-cycle cost. Yet, major design decisions are made, the results of which end up in the system specification, and all subsequent design activity must comply. Thus, the need for a strong system engineering thrust at this early stage in the life cycle is critical.

2.4 SYSTEM OPERATIONAL REQUIREMENTS

With the identification of a need, combined with the selection of a feasible technical design approach, it is necessary to project this information in terms of anticipated operational requirements. Operational requirements reflect the needs of the consumer relative to system utilization and the accomplishment of a mission. The operational concept, as defined herein, includes the following information:

1. *Operational distribution or deployment:* The number of consumer sites where the system will be utilized, the geographical distribution and schedule, and the type and quantity of system components at each location. These factors respond to the question, Where is the system to be utilized? Figure 2.2 presents a sample worldwide distribution scheme.

2. *Mission profile or scenario:* Description of the prime mission of the system and its alternative or secondary missions. What is the system to accomplish in responding to the need? How will it accomplish its objective? The response to these questions may be defined through the development of a series of operational profiles, illustrating the dynamic aspects required in accomplishing a mission. An aircraft flight path between two cities, an automobile route, and a shipping route are examples. Figure 2.3 provides a simple illustration of possible profiles.

3. *Performance and related parameters:* Definition of the basic operating characteristics, or functions, of the system. This refers to parameters such as range, accuracy, rate, capacity, throughout, power output, size, and weight. What are the critical system performance parameters necessary to accomplish the mission at the various consumer sites? These should be related to the profiles in Figure 2.3.

4. *Utilization requirements:* Anticipated usage of the system and its components, in accomplishing its mission. This refers to hours of system operation per day, duty cycle, on–off cycles per month, percentage of total capacity utilized, facility loading, and so on. How are the various system components to be utilized? This investigation leads to determining some of the stresses imposed on the system by the operator and its environment.

5. *Effectiveness requirements:* System requirements, specified quantitatively as applicable, including cost/system effectiveness, operational availability, dependability, reliability mean time between failure (MTBF), failure rate (λ), readiness rate,

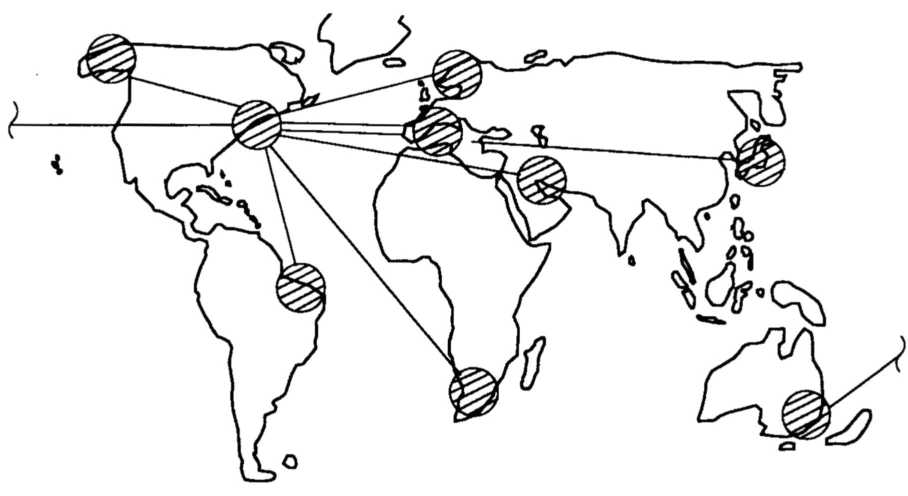


Figure 2.2 System operational requirements (geographical distribution).

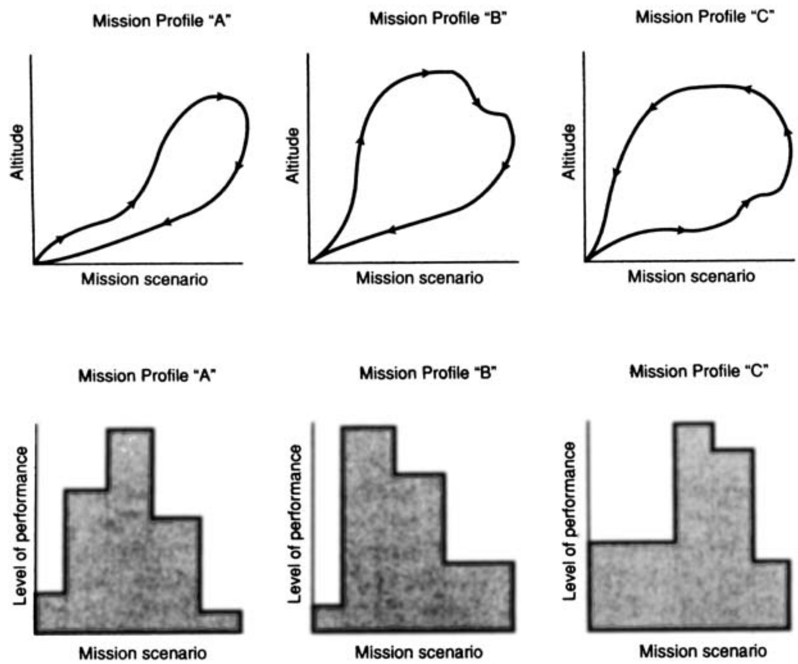


Figure 2.3 Sample system operational profiles. *Source:* LOGISTICS ENGINEERING AND MANAGEMENT 5/E by Blanchard, Benjamin S., © Reprinted by permission of Pearson Education, Inc., Upper Saddle River, NJ.

maintenance downtime (MDT), mean time between maintenance (MTBM), facility utilization (in percent), personnel skill levels, cost, and so on. Given that the system will perform, how *effective* or *efficient* will it be?

6. *Operational life cycle (horizon):* The anticipated time that the system will be in operational use. How long will the system be in use by the consumer? What is the total inventory profile for the system and its components, and where is this inventory to be located? The anticipated system life cycle must be defined.

7. *Environment:* Definition of the environment in which the system is expected to operate in an effective manner; for example, temperature, shock and vibration, noise, humidity, arctic or tropics, mountainous or flat terrain, airborne, ground, or shipboard. Following a set of mission profiles may result in specifying a range of values. To what will the system be subjected during its operational use, and for how long? In addition to system operations, environmental considerations should address transportation, handling, and storage modes. It is possible that the system (and/or some of its components) will be subjected to a more rigorous environment during transportation than during its operation.

The establishment of operational requirements forms the basis for system design. Obviously, one needs answers to the following questions before proceeding further:

1. *What* function(s) will the system perform?
2. *When* will the system be required to perform its intended function?
3. *Where* will the system be utilized and for how long?
4. *How* will the system accomplish its objective?

In responding to these questions, a baseline must be established. Although conditions may change, some initial assumptions are required. For example, system components will be utilized differently at different consumer locations, the distribution of system components may vary as the need changes, and/or the length of the life cycle may change as a result of obsolescence or the effects of competition. Nevertheless, the aforementioned information has to be developed in order to proceed with system design.

In the past, the operational requirements for many new systems were developed (by a consultant, a marketing group, or an equivalent organizational entity), placed in a file while awaiting a decision to proceed with preliminary design, and then forgotten when subsequent design activity finally resumed. At that point, with the need for this type of information readily apparent (but the information unavailable), individual design groups generated their own assumptions. Moreover, not all of the design functions were referencing the same baseline, and conflicting requirements evolved. This, in turn, led to the development of systems that did not meet consumer requirements and the initiation of corrective action through costly modifications. In other words, if the applicable operational requirements are not well defined and integrated into the design process, the later results can be quite costly.

This is another critical area of activity where a strong system engineering thrust is necessary. The operational requirements for the system must be thoroughly defined and integrated, and the appropriate information must be disseminated in a timely manner throughout all applicable design organizations. Everyone involved in the design process must track the same baseline.

2.5 THE MAINTENANCE AND SUPPORT CONCEPT

In addressing system requirements, the normal tendency is to deal primarily with those elements of the system that relate directly to the "performance of the mission"; that is, prime equipment, operator personnel, operational software, and associated data. At the same time, there is very little attention given to system support. In general, the emphasis in the past has been directed toward only *part* of the system, and not the *entire* system. This, of course, has led to some of the problems discussed in Section 1.1.

To meet the overall objectives of system engineering, it is essential that *all* aspects of the system be considered on an integrated basis. This includes not only the prime mission-oriented segments of the system, but the support capability as well. System support must be considered from the beginning (e.g., during the feasibility analysis when new technologies are being evaluated for possible application), and a before-the-fact *maintenance concept* must be developed as to how the proposed system is to be supported on a life-cycle basis.[3]

The maintenance concept, developed during conceptual design, evolves from the definition of system operational requirements, as illustrated by the flowchart in Figure 2.4. Initially, one must deal with the flow of activities and materials from design, through production, and to the consumer's operational site(s) where the system is being utilized. In addition, there is a flow involving the system support capability. In Figure 2.4, a maintenance flow exists when items are returned from the operational site to the intermediate and depot levels of maintenance. A second flow involves the distribution of spare parts, personnel, test equipment, and data from the various suppliers to the intermediate and depot levels of maintenance, and to the operational sites as required. The flowchart in Figure 2.4 reflects the activities that are related to the overall system support capability.

Although there are some variations as a function of the nature and type of system, the maintenance concept generally includes the following information:

1. *Levels of maintenance:* Corrective and preventive maintenance may be performed on the system itself (or an element thereof) at the site where the system is used by the consumer, in an intermediate shop near the consumer, and/or at a depot or manufacturer's facility. Maintenance level pertains to the division of functions and tasks for each area where maintenance is performed. Anticipated frequency of maintenance, task complexity, personnel skill-level requirements, special facility needs, and so on, dictate to a great extent the specific functions to be accomplished at each level. Depending on the nature and mission of the system, there may be two, three, or four levels of maintenance. However, for the purposes of further discussion, maintenance is classified as "organizational," "intermediate," and "supplier/depot."

a. *Organizational maintenance.* Organizational maintenance is performed at the operational site (e.g., airplane, vehicle, manufacturing production line, or communication facility). Generally, it includes tasks performed by the using organization on its own equipment. Organizational-level personnel are usually involved with the operation and use of equipment and have minimum time available for detail system maintenance. Maintenance at this level normally is limited to periodic checks of equipment performance, visual inspections, cleaning of system elements, verification of software, some servicing, external adjustments, and the removal and replacement of some components. Personnel assigned to this level generally do not repair the removed components, but forward them to the intermediate

[3]The *maintenance concept* is defined in this text as a "before-the-fact series of illustrations and statements pertaining to how the system is to be designed for supportability (e.g., two versus three levels of maintenance, system/component packaging, degree of diagnostics incorporated, quantitative effectiveness requirements for the various elements of support, etc.)," whereas the *maintenance plan* defines the specific requirements for support based on a known configuration and on the results from a supportability analysis. The maintenance concept is an *input* to design, and the maintenance plan is the *result* of design. Complete coverage of the maintenance concept is presented B. S. Blanchard, *Logistics Engineering and Management,* 5th ed. (Upper Saddle River, NJ: Prentice-Hall), 1998.

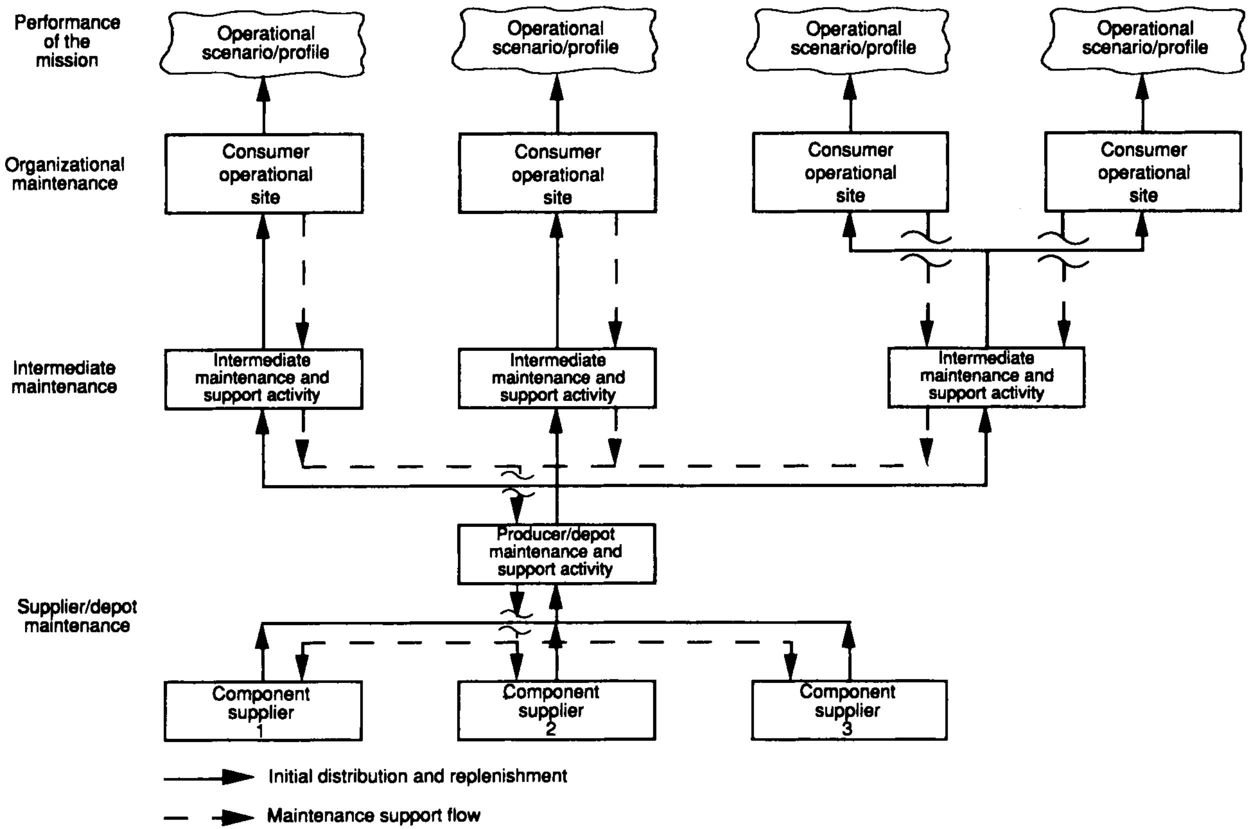


Figure 2.4 System operational and maintenance flow.

level. From the standpoint of maintenance, the least skilled personnel are assigned to this function. The design of equipment must take this fact into consideration (i.e., design for simplicity).

b. *Intermediate maintenance.* Intermediate maintenance tasks are performed by mobile, semimobile, and/or fixed specialized organizations and installations. At this level, end items may be repaired by the removal and replacement of major modules, assemblies, or piece parts. Scheduled maintenance requiring equipment disassembly may also be accomplished. Available maintenance personnel are usually more skilled and better equipped than those at the organizational level and are responsible for performing more detail maintenance.

Mobile or semimobile units are often assigned to provide close support to deployed operational systems. These units may be vans, trucks, or portable shelters containing some test and support equipment and spares. The mission is to provide on-site maintenance (beyond that accomplished by organizational-level personnel) to facilitate the return of the system to its full operational status on an expedited basis. A mobile unit may be used to support more than one operational site. A good example is the maintenance vehicle that is deployed from the airport hangar to an airplane parked at a commercial airline terminal gate and needing extended maintenance.

Fixed installations (permanent shops) are generally established to support both the organizational-level tasks and the mobile or semimobile units. Maintenance tasks that cannot be performed by the lower levels, due to limited personnel skills and test equipment, are performed here. High personnel skills, additional test and support equipment, more spares, and better facilities often enable equipment repair to the module and component part level. Fixed shops are usually located within specified geographical areas.

Rapid maintenance turnaround times are not as imperative here as at the lower levels of maintenance.

c. *Depot or supplier maintenance.* The depot level constitutes the highest type of maintenance and supports the accomplishment of tasks above and beyond the capabilities available at the intermediate level. Physically, the depot may be a specialized repair facility supporting a number of systems/equipment in the inventory or may be the equipment manufacturer's plant. Depot facilities are fixed, and mobility is not a problem. Complex and bulky equipment, large quantities of spares, environmental control provisions, and so on, can be provided if required. The high-volume potential in depot facilities fosters the use of assembly-line techniques, which, in turn, permits the use of relatively unskilled labor for a large portion of the workload, with a concentration of highly skilled specialists in such certain key areas as fault diagnosis and quality control.

The depot level of maintenance includes the complete overhauling, rebuilding, and calibration of equipment, as well as the performance of highly complex maintenance actions. In addition, the depot provides an inventory supply capability. The depot facilities are generally remotely located to support the needs of a specific geographical area or designated product lines.

The three levels of maintenance are presented in Figure 2.5.[4]

2. *Repair policies:* Within the constraints illustrated in Figures 2.4 and 2.5, there may be a number of possible policies specifying the extent to which repair of a system component will be accomplished (if at all). A repair policy may dictate that an item should be designed to be nonrepairable, partially repairable, or fully repairable. Repair policies are established initially, criteria are then developed, and system design progresses within the bounds of the repair policy that is selected. An example of a repair policy, for System XYZ, developed as part of the maintenance concept during conceptual design, is illustrated in Figure 2.6.[5]

3. *Organizational responsibilities:* The accomplishment of maintenance may be the responsibility of the consumer, the producer (or supplier), a third party, or a combination thereof. In addition, the responsibilities may vary, not only with different components of the system, but over time, through operational use of the system and the sustaining support phase. Decisions pertaining to organizational responsibilities may impact system design from a diagnostic and packaging standpoint, as well as dictate repair policies, contract warranty provisions, and the like. Although conditions may change, some initial assumptions are required at this point.

4. *Maintenance support elements:* As part of the initial maintenance concept, criteria must be established relating to the various elements of maintenance support. These elements include supply support (spare and repair parts, associated inventories, provisioning data), test and support equipment, personnel and training, transportation and handling equipment, facilities, data, and computer resources. Such criteria, as an input to design, may cover self-test provisions, built-in versus external test requirements, packaging and standardization factors, personnel quantities and skill levels, transportation and handling factors and constraints, and so on. The maintenance concept provides some initial system design criteria pertaining to the activities illustrated in Figure 2.4, and the final determination of specific logistic and maintenance support requirements will occur through the completion of a maintenance engineering analysis as design progresses.

5. *Effectiveness requirements:* These requirements are the factors associated with the support capability. In the supply support area, they may include a spare part demand rate; the probability of a spare part being available when required; the probability of mission success given a designated quantity of spares in the inventory; and the economic order quantity as related to inventory procurement. For test equipment, the length of the queue while waiting for test, the test station process time, and the

[4]The criteria presented in the figure represent just an example of the guidance information that might be established as a start in attempting to develop various maintenance policies (or the extent of maintenance to be accomplished at each level). Actually, one will need to consider a number of factors (e.g., economics, technology, social and security factors, warranty provisions, cost, and component criticality) in determining what will be repaired and where, and these need to be tailored to the particular system being addressed.

[5]Repair policies are usually verified through a *level-of-repair analysis* (LORA), initially accomplished in conjunction with the maintenance concept development, later accomplished as part of a maintainability analysis and/or supportability analysis, and ultimately leading to the development of the maintenance plan. Refer to Chapter 3 and the case study presented in Appendix B for additional discussion.

Criteria	Organizational Maintenance	Intermediate Maintenance		Depot Maintenance
Done where?	At the operational site or wherever the prime equipment is located	Mobile or semimobile units	Fixed units	Depot facility
		Truck, van, portable shelter, or equivalent	Fixed field shop	Specialized repair activity, or manufacturer's plant
Done by whom?	System/equipment operating personnel (low maint. skills)	Personnel assigned to mobile, semimobile, or fixed units (intermediate maintenance skills)		Depot facility personnel or manufacturer's production personnel (mix of intermediate production personel skills and high maintenance skills)
On whose equipment?	Using organization's equipment	Equipment owned by using organization		
Type of work accomplished?	Visual inspection Operational checkout Minor servicing External adjustments Removal and replacement of some components	Detailed inspection and system checkout Major servicing Major equipment repair and modifications Complicated adjustments Simple software maintenance Limited calibration Overload from organizational level of maintenance		Complicated factory adjustments Complex equipments repairs and modifications Overhaul and rebuild Detailed calibration Software maintenance (detailed modifications Supply support Overload from intermediate level of maintenance

Figure 2.5 Major levels of maintenance.

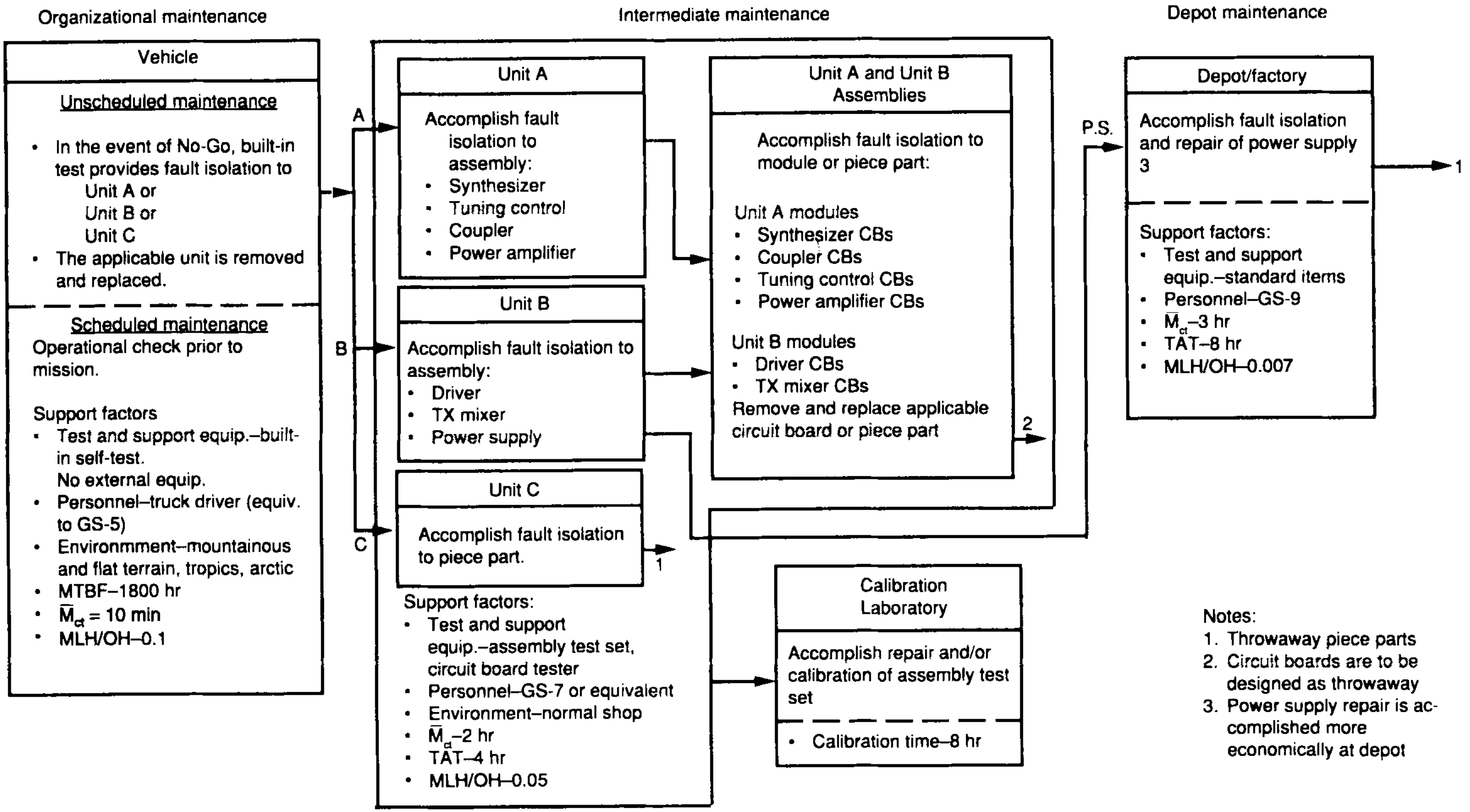


Figure 2.6 System maintenance concept flow (repair policy.) *Source:* LOGISTICS ENGINEERING AND MANAGEMENT 5/E by Blanchard, Benjamin S., © Reprinted with permission of Pearson Education, Inc., Upper Saddle River, NJ.

test equipment reliability are key factors. In transportation, transportation rates, transportation times, the reliability of the transportation, and transportation costs are of significance. For personnel and training, one should be interested in personnel quantities and skill levels, human error rates, training rates, training times, and training equipment reliability. In software, the number of errors per mission segment or per line of code may be important measures. These factors, as related to a specific system-level requirement, must be addressed. It is meaningless to specify a tight quantitative requirement applicable to the repair of a prime element of the system when it takes six months to acquire a needed spare part. The effectiveness requirements applicable to the support capability must complement the requirements of the system overall.

6. *Environment:* Definition of the environment as it pertains to maintenance and support. This includes temperature, shock and vibration, humidity, noise, arctic versus tropical environment, mountainous versus flat terrain, shipboard versus ground conditions, and so on, as applicable to maintenance activities and related transportation, handling, and storage functions.

In summary, the maintenance concept provides the basis for the establishment of supportability requirements in system design. Not only do these requirements impact the prime mission-oriented segments of the system, but they should provide guidance in the design and/or procurement of the necessary elements of logistic support. In addition, the maintenance concept forms the baseline for the development of the detailed maintenance plan, prepared during the detail design and development phase.[6]

2.6 IDENTIFICATION AND PRIORITIZATION OF TECHNICAL PERFORMANCE MEASURES (TPMS)

With the development of the operational requirements and the maintenance concept for the system, it is necessary for the designer to review these requirements in terms of relative degrees of importance, criticality from the standpoint of accomplishing the desired mission(s), and priorities in design in the event that trade-offs are necessary. In the design of a vehicle, is *speed* more important than *size?* For a manufacturing plant, is *production quantity* more important than *product quality?* In a communication system, is *range* more important than *reliability* or *clarity of message?* For a computer capability, is *capacity* more important than *speed?*

The number of objectives may be numerous, and the designer needs to understand which are more important than others and the relationships between them. In addition, it is desirable to express these objectives in quantitative terms where feasible. It is difficult (if not impossible) to proceed with the design in a satisfactory manner unless there are some "measurable" goals specified from the beginning. These goals, in turn, must reflect the customer's (consumer's) requirements.

[6]It should be noted that development of system operational requirements and the maintenance and support concept should (together) cover all of the activities shown with the *forward* and *reverse* flows in Figure 1.20.

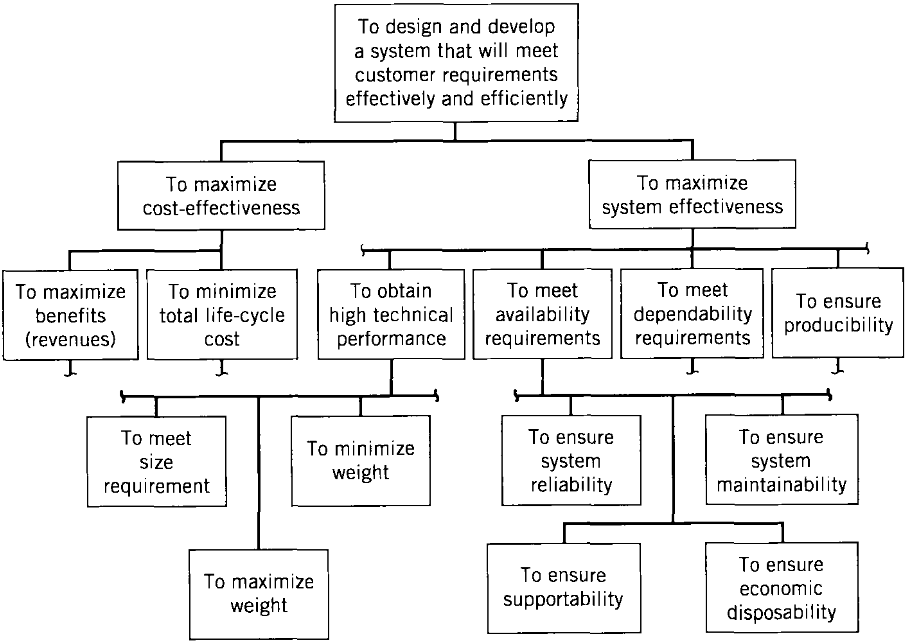


Figure 2.7 Objectives tree (partial).

In the development of a system's operational requirements and the maintenance concept, there are a number of measurable goals. The use of an "objectives tree," or something of an equivalent nature, may aid in facilitating prioritization. As shown in Figure 2.7, requirements are often expressed in very general qualitative terms and included in a specification. The question is, How does one respond to a requirement such as "The system must be designed to meet customer requirements effectively and efficiently"? How does one measure the results for the purposes of validation?

In the absence of better guidance, the designer will need to interpret the specified requirements and make some assumptions relative to what is meant by "effectively" and "efficiently." Although the objective is to design a system in response to consumer requirements, it may not always happen unless there is a good communications link between the designer and the customer. Through a team effort, the approach conveyed in Figure 2.7 can help clarify the requirements. Initially, it may be necessary to express design objectives in qualitative terms, showing their relationships in a top-down hierarchical manner. Subsequently, an attempt should be made to establish *quantitative* measures for each block in the figure and ensure that the appropriate "traceability" exists both downward and upward. Applying this approach to the system breakdown in Figure 1.14, what measures should be applied and to what level in the overall hierarchical structure for the system? Further, what design criteria should be established for each level? Is *reliability* more important than *maintainability?* Are *human factors* more important than *cost?* Establishing these relationships will, in turn, help the designer to identify areas where emphasis must be applied in the design process and the areas that can be traded-off in the event that something has to "give."

An excellent tool that can be applied to aid in establishing the necessary communications between designers and the consumer (i.e., the "customer") is the *quality function deployment* (QFD) method. QFD constitutes a "team" approach to help ensure that the voice of the customer is reflected in the ultimate design. The purpose is to establish the necessary requirements and to translate those requirements into technical solutions. Consumer requirements and preferences are defined and categorized as *attributes*, which are then weighted according to the degree of importance. The QFD method gives the design team an understanding of customer desires, forces the customer to prioritize those desires, and enables a comparison of one design approach against another. Each customer attribute is then satisfied by a technical solution.[7]

The QFD process involves constructing one or more matrices, the first of which is often referred to as the "House of Quality" (HOQ). A modified version of the HOQ is presented in Figure 2.8. Starting on the left side of the structure is the identification of customer needs and the ranking of those needs in terms of priority, the levels of importance being specified quantitatively. This reflects the "WHATs" that must be addressed. A team, with representation from both consumer and design organizations, determines the priorities through an iterative process of review, evaluation, revision, reevaluation, and so on. The top part of the HOQ identifies the designer's *technical* response relative to the attributes that must be incorporated into the design in order to respond to the needs (i.e., the "voice of the customer"). This constitutes the "HOWs," and there should be at least one technical solution for each identified customer need. The interrelationships among attributes (or technical correlations) are identified, as well as possible areas of conflict. The center part of the HOQ conveys the strength of the proposed technical response or its impact on the identified requirement. The bottom part allows for a comparison between possible alternatives, and the right side of the HOQ is used for planning purposes.[8]

The QFD method is used to facilitate the translation of a prioritized set of subjective customer requirements into a set of *system-level* requirements during conceptual design. A similar approach may be used to subsequently translate system-level requirements into a more detailed set of requirements at each stage in the design and development process. In Figure 2.9, the "HOWs" from one house become the "WHATs" for a succeeding house. Requirements may be developed for the system, subsystem, component, the manufacturing process, the support infrastructure, and so on. The objective is to ensure the required justification and traceability of requirements from the top down. Further, requirements should be stated in *functional* terms.

Although the QFD method may not be the only approach used in defining the requirements for system design, it does constitute an excellent tool for creating the necessary visibility from the beginning. One of the largest contributors to risk is the lack of a good set of requirements and an adequate system specification. Inherent within

[7]Three good references pertaining to the QFD process are (a) L. Cohen, *Quality Function Deployment: How to Make QFD Work for You* (Reading, MA: Addison-Wesley, 1995); (b) J. B. Revelle, J. W. Moran, and C. Cox, *The QFD Handbook* (New York: John Wiley & Sons, Inc., 1997); and (c) P. Biren, "Review of QFD and Related Deployment Techniques," *Journal of Manufacturing Systems* 17, no. 3 (1998).
[8]J. R. Hauser, and D. Clausing, "The House of Quality," *Harvard Business Review*, (May–June 1988).

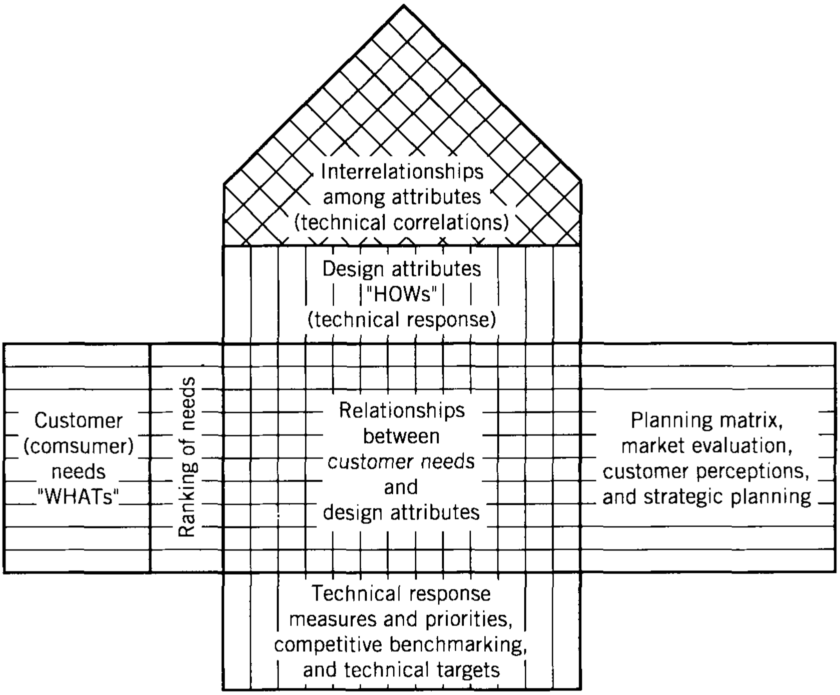


Figure 2.8 House of Quality (modified).

the system specification should be the identification and prioritization of technical performance measures (TPMs), as illustrated in Figure 2.10. The TPM, its associated measure (i.e., "metric"), its relative importance, and "benchmark" objective in terms of what is currently available will provide designers with the necessary guidance for accomplishing their task. This is essential for establishing the appropriate levels of design emphasis, for defining the criteria as an input to the design, and for identifying the levels of possible risk should the requirements not be met.

2.7 FUNCTIONAL ANALYSIS

An essential element of early conceptual and preliminary design is the development of a *functional* description of the system to serve as a basis for the identification of the resources necessary for the system to accomplish its objective(s). A function is a specific or discrete action (or series of actions) necessary to achieve a given objective; that is, an operation that the system must perform to accomplish its mission, or a maintenance action that is necessary to restore the system to operational use. Such actions may ultimately be accomplished through the use of equipment, people, software, facilities, data, or combinations thereof. However, at this point, the objective is to specify the "WHATs" and *not* the "HOWs"; that is, *what* needs to be accomplished

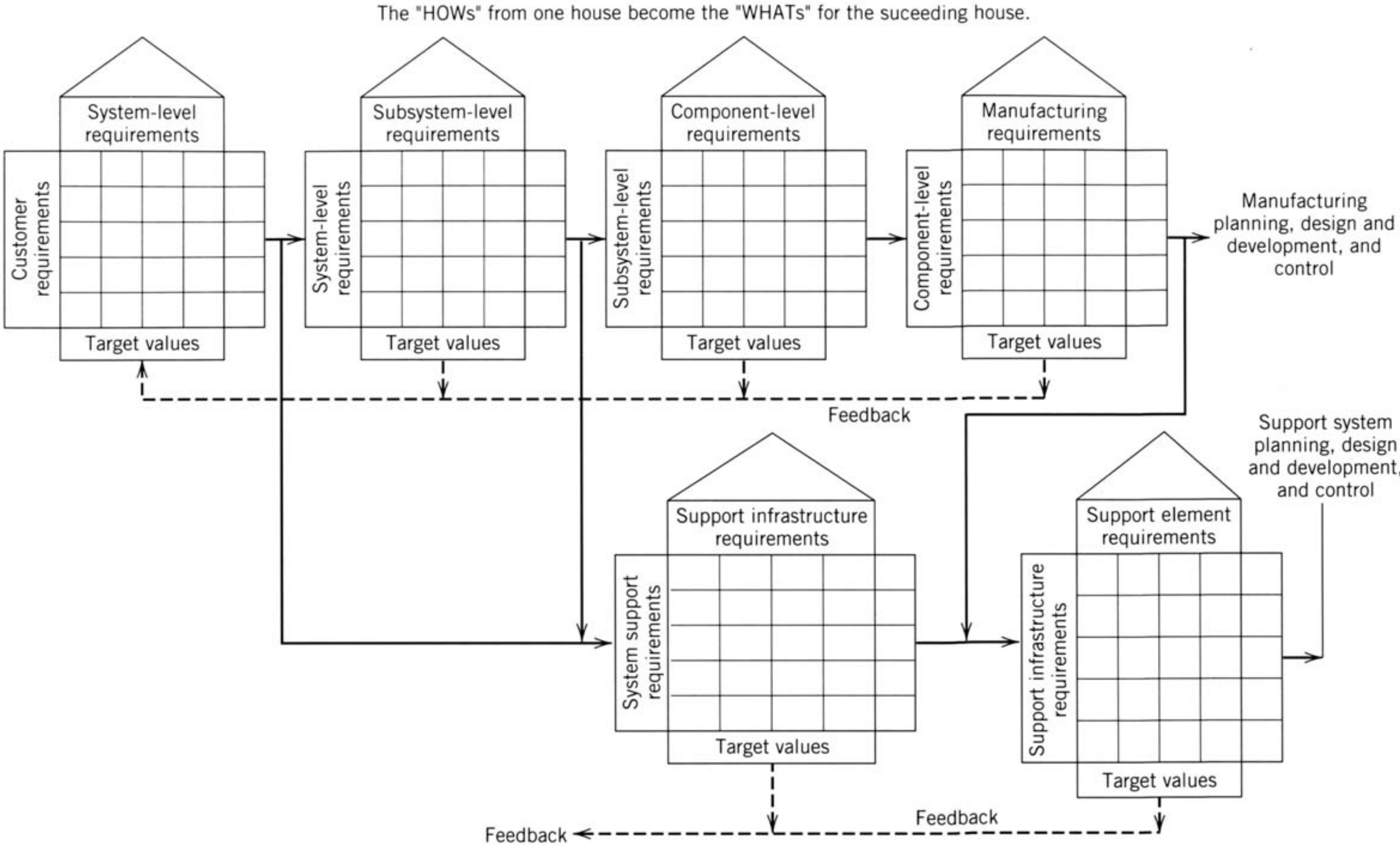


Figure 2.9 Family of houses (traceability of requirements).

Technical performance measure (TPM)	Quantative requirement ("metric")	Current "benchmark" (competing systems)	Relative importance (customer desires)
Process time (days)	30 days (maximum)	45 days (system "M")	10
Velocity (mph)	100 mph (minimum)	115 mph (system "B")	32
Availability (operational)	98.5% (minimum)	98.9% (system "H")	21
Size (feet)	10 feet long 6 feet wide 4 feet high (maximum)	9 feet long 8 feet wide 4 feet high (system "M")	17
Human factors	Less than 1% error rate per year	2% per year (system "B")	5
Weight (pounds)	600 pounds (maximum)	650 pounds (system "H")	6
Maintainability (MTBM)	300 miles (minimum)	275 miles (system "H")	9
			100%

Figure 2.10 Prioritization of technical performance measures (TPMs).

versus *how* it is to be done. The functional analysis is an iterative process of breaking down requirements from the system level, to the subsystem, and as far down the hierarchical structure as necessary to identify input design criteria and/or constraints for the various elements of the system.[9]

In Figure 2.1, the functional analysis may be initiated in the early stages of conceptual design as part of the problem definition and needs analysis task, and functions that the system must perform in order to fulfill the needs of the consumer are identified. These *operating* functions are then expanded and formalized through the development of system operational requirements. Primary *maintenance and support* functions for the system, which evolve from the operational requirements, are identified as part of the maintenance concept development process. Subsequently, these functions must be expanded to include *all* of the activity, from the initial identification of need to the retirement of the system.

[9]In applying the principles of system engineering, not one piece of equipment, or element of software, or data item, or element of support should be identified and purchased without the need for such having first been justified through a functional analysis. On many projects, items are often purchased on the basis of what was initially perceived as a "requirement," but which later turned out not to be needed in the end. This practice, of course, can turn out to be quite costly.

A functional analysis can be facilitated through the use of functional flow block diagrams, as illustrated in Figure 2.11. Block diagrams are developed primarily for the purpose of structuring system requirements into "functional terms." They are developed to illustrate basic system organization and to identify functional interfaces. The functional analysis (and the generation of functional flow diagrams) is intended to enable the completion of the design, development, and system definition process in a comprehensive and logical manner. Top-level requirements are identified, partitioned to a second level, and on down to the depth required for the purposes of definition. More specifically, the functional approach helps to ensure the following:[10]

1. That all facets of system design and development, production, operation, support, and retirement are covered; that is, all significant activities within the system life cycle
2. That all elements of the system are fully recognized and defined; that is, prime equipment, spare/repair parts, test and support equipment, facilities, personnel, data, and software
3. That a means is provided for relating system packaging concepts and support requirements to specific system functions; that is, satisfying the requirements of good *functional* design
4. That the proper sequences of activity and design relationships are established, along with critical design interfaces

One of the objectives of functional analysis is to ensure traceability from the top system-level requirements down to the requirements for detail design. In Figure 2.12, it is assumed that there is a need for transportation between City "A" and City "B." Through a feasibility analysis, trade-off studies were accomplished, and the results indicate that transportation by air is the preferred mode. Subsequently, through the definition of operational requirements, it was concluded that there is a requirement for a new aircraft system, demonstrating good performance and effectiveness characteristics, with quantitative goals specified for size, weight, thrust, range, fuel capacity, reliability, maintainability, supportability, cost, and so on. An aircraft must be designed and produced that will accomplish its mission in a satisfactory manner, flying through a number of operational profiles such as the one illustrated in Figure 2.12. Further, the maintenance concept indicates that the aircraft will be designed for support at three levels of maintenance by the user, will incorporate built-in test provisions, and will be in operational use for a life cycle of 10 years.

With this basic information, following the general steps in Figure 2.1, one can commence with the structuring of the system in functional terms. A top-level functional flow diagram can be developed to cover the primary activities identified within the specified life cycle. Each of these designated activities can be expanded through

[10]The preparation of functional flow block diagrams (FFBDs) may be accomplished through the use of any one of a number of graphical methods, including the Integrated DEFinition (IDEF) modeling method, the Behavioral Diagram method, and the N-Squared Charting method. Although the graphical descriptions are different, the ultimate objectives are similar.

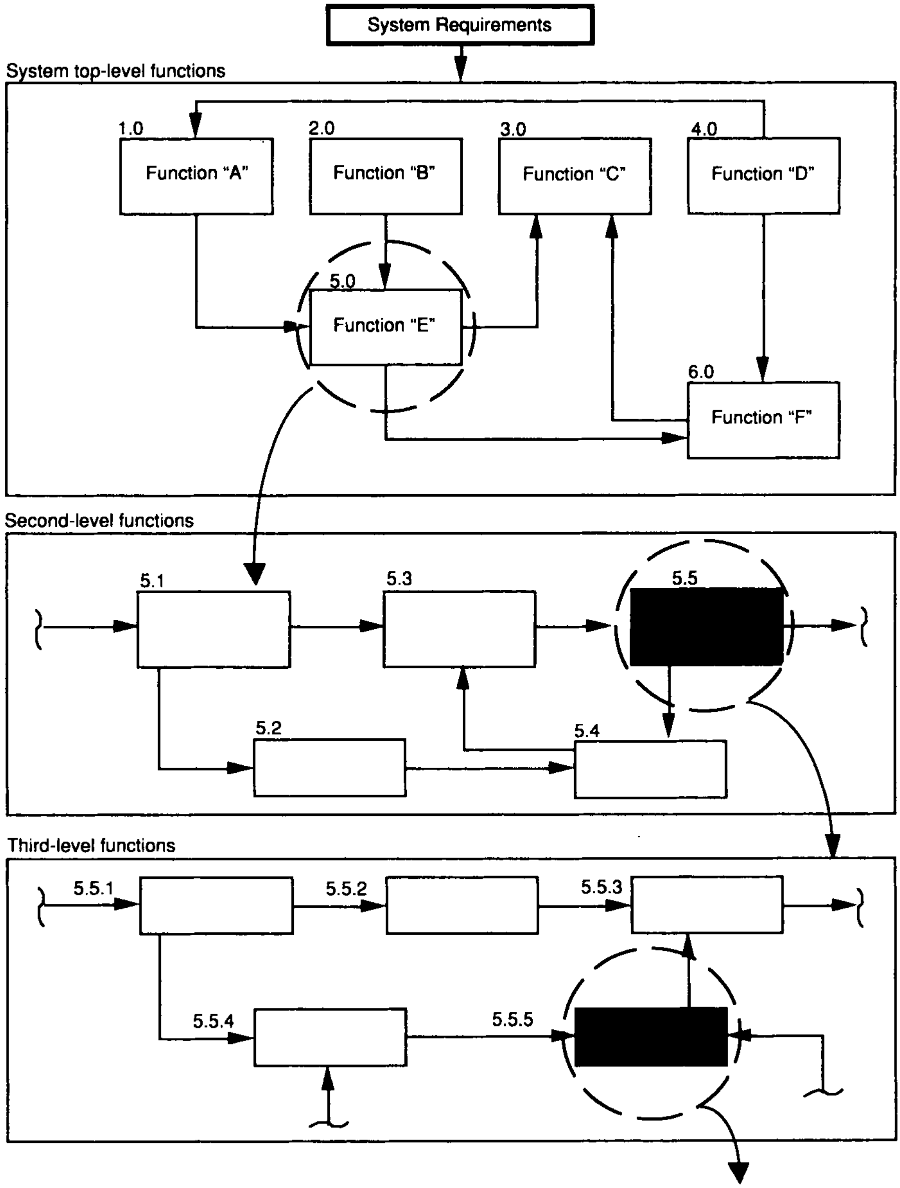


Figure 2.11 System functional breakdown.

a second-level functional flow diagram, a second-level activity into a third-level functional flow, and so on.

Through this progressive expansion of functional activities, directed to defining the "WHATs" (versus the "HOWs"), one can evolve from the mission profile in Figure 2.12 down to a specific aircraft capability such as "communications." A commu-

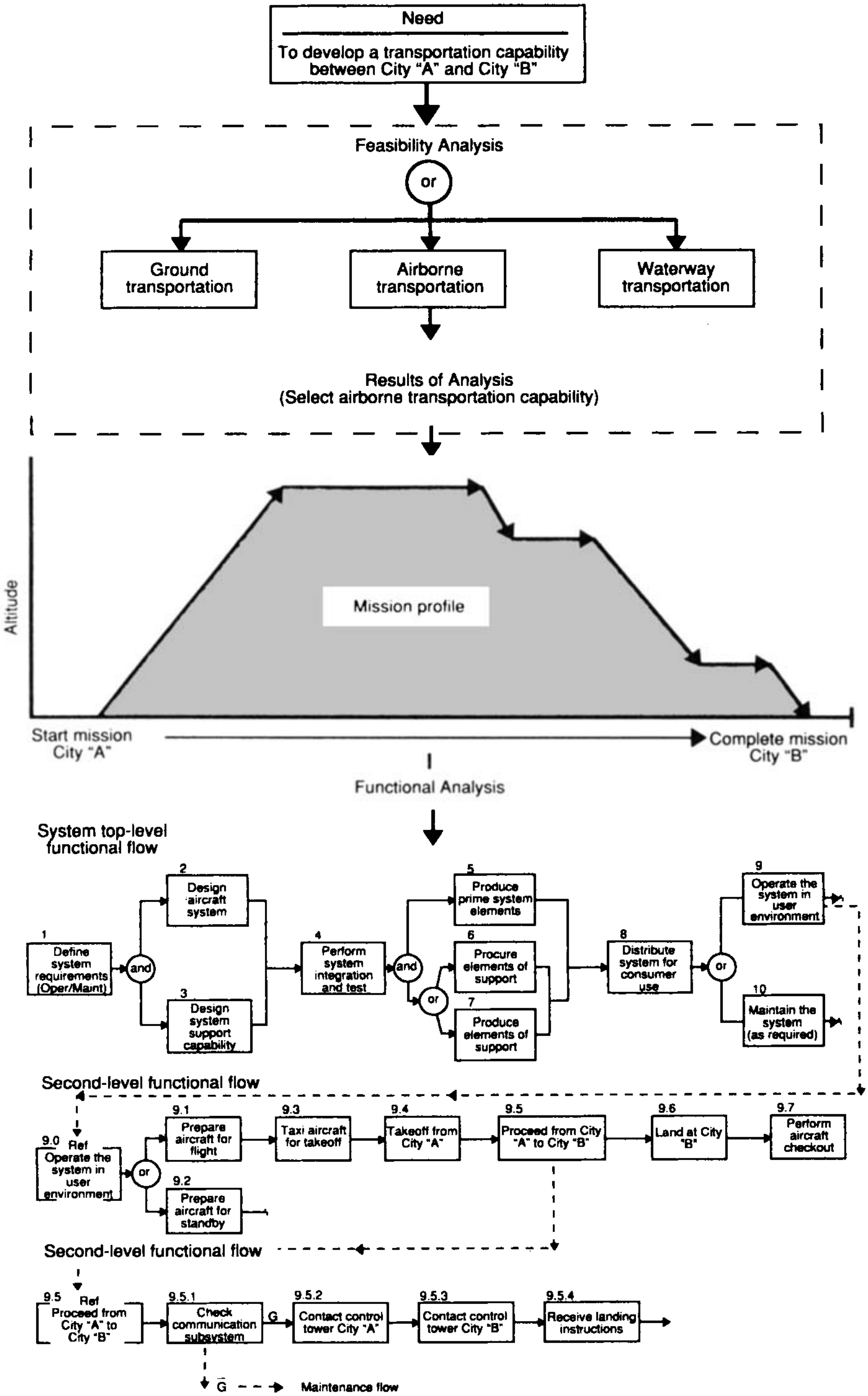


Figure 2.12 Evolutionary development of functional requirements.

nications subsystem is identified, trade-offs are accomplished, and a detail design approach is selected. Specific resources that are necessary to respond to the stated functional requirement can be identified. In other words, one can drive downward from the system level to identify the resources needed to perform certain functions (e.g., equipment, people, facilities, and data). Also, given a specific equipment requirement, one can progress "upward" for *justification* of that requirement. The functional analysis provides the mechanism for "down–up" traceability.

2.7.1 Functional Flow Block Diagrams (FFBDs)

In the development of functional flow diagrams, some degree of standardization is necessary, for the purpose of "communication," in defining the system. Thus, certain basic practices and symbols should be used, whenever possible, in the physical layout of functional diagrams. The following paragraphs provide some guidance in this direction:

1. *Function block:* Each separate function in a functional diagram should be presented in a single box enclosed by a solid line. Blocks used for reference to other flows should be indicated as partially enclosed boxes labeled "REF." Each function may be as gross or detailed as required by the level of the functional diagram on which it appears, but it should stand for a definite, finite, discrete action to be accomplished by equipment, personnel, facilities, software, or any combination thereof. Questionable or tentative functions should be enclosed in dotted blocks.

2. *Function numbering:* Functions identified on the functional flow diagrams at each level should be numbered in a manner that preserves the continuity of functions and provides information with respect to function origin throughout the system. Functions in the top-level functional diagram should be numbered 1.0, 2.0, 3.0, and so on. Functions that further indenture these top functions should contain the same parent identifier and should be coded at the next decimal level for each indenture. For example, the first indenture of function 3.0 would be 3.1, the second 3.1.1, the third 3.1.1.1, and so on. For expansion of a higher-level function within a particular level of indenture, a numerical sequence should be used to preserve the continuity of function. For example, if more than one function is required to amplify function 3.0 at the first level of indenture, the sequence should be 3.1, 3.2, 3.3, . . . , 3.*n*. For expansion of function 3.3 at the second level, the numbering will be 3.3.1, 3.3.2, . . . , 3.3.*n*. Where several levels of indentures appear on a single functional diagram, the same pattern should be maintained. Whereas the basic ground rule should be to maintain a minimum level of indentures on any one particular flow, it may become necessary to include several levels to preserve the continuity of functions and to minimize the number of flows required to functionally depict the system.

3. *Functional reference:* Each functional diagram should contain a reference to its next higher functional diagram through the use of a reference block. For example, function 4.3 should be shown as a reference block in the case where functions 4.3.1, 4.3.2, . . . , 4.3.*n*, are being used to expand function 4.3. Reference blocks should also be used to indicate interfacing functions as appropriate.

4. *Flow connection:* Lines connecting functions should indicate only the functional flow and should not represent either a lapse in time or any intermediate activity. Vertical and horizontal lines between blocks should indicate that all functions so interrelated must be performed in either a parallel or a series sequence. Diagonal lines may be used to indicate alternative sequences (cases where alternative paths lead to the next function in the sequence).

5. *Flow directions:* Functional diagrams should be laid out so that the functional flow is generally from left to right, and the reverse flow, in the case of a feedback functional loop, from right to left. Primary input lines should enter the function block from the left side; the primary output, or *GO* line, should exit from the right; and the *NO-GO* line should exit from the bottom of the box.

6. *Summing gates:* A circle should be used to depict a summing gate. As in the case of functional blocks, lines should enter and/or exit the summing gate as appropriate. The summing gate is used to indicate convergence or divergence, or parallel or alternative functional paths, and is annotated with the term AND or OR. The term AND is used to indicate that parallel functions leading into the gate must be accomplished before proceeding to the next function, or that paths emerging from the AND gate must be accomplished after the preceding functions. The term OR is used to indicate that any of several alternative paths (alternative functions) converge to, or diverge from, the OR gate. The OR gate thus indicates that alternative paths may lead or follow a particular function.

7. *Go and no-go paths:* The symbols G and $\bar{G}$ are used to indicate go and no-go paths, respectively. The symbols are entered adjacent to the lines leaving a particular function to indicate alternative functional paths.

8. *Numbering procedure for changes to functional diagrams:* Additions of functions to existing data should be accomplished by locating a new function in its correct position without regard to sequence of numbering. The new function should be numbered using the first unused number at the level of indenture appropriate for the new function.

The functions identified should not be limited strictly to those necessary for the operation of the system, but must consider the possible effects of maintenance on system design. In most instances, maintenance functional flows will evolve directly from operational flows.

2.7.2 Operational Functions

Operational functions, in this instance, constitute those that describe the activities that must be accomplished in order to fulfill the mission requirements. These may include both (1) those activities that involve the design, development, production, and distribution of a system for use and (2) those activities that are related directly to the completion of a consumer mission scenario. In the second category, these may include a description of the various modes of system operation and utilization. For instance, typical gross operating functions may entail (1) "prepare aircraft for flight,"

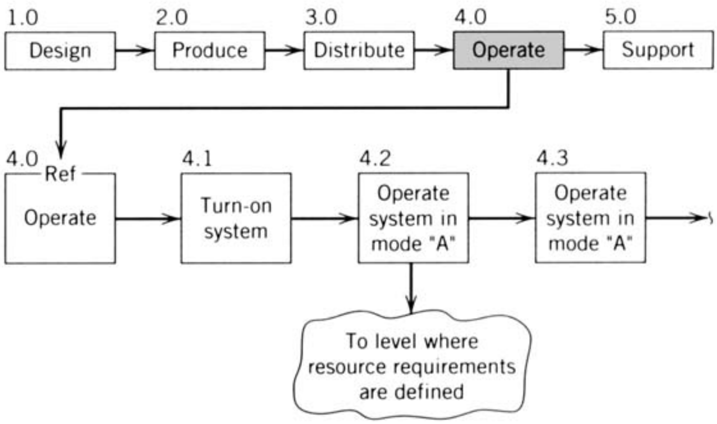


Figure 2.13 Functional block diagram (partial).

(2) "transport material from the factory to the warehouse," (3) "initiate communications between the producer and the user," (4) "produce x quantity of units in a seven-day time frame," and (5) "process a data to eight company distribution outlets, in b time, with c accuracy, and in d format." System functions necessary to successfully complete the identified modes of operation are then described.

Figure 2.13 illustrates a simplified operational flow diagram. Note that the words in each block are "action oriented" and the block numbering allows for the downward–upward traceability of resource requirements. The functions are broken down to the depth necessary to describe the resources that will be required to accomplish the function—that is, equipment, software, people, facilities, and so on.

2.7.3 Maintenance and Support Functions

Once operational functions are described, the system development process leads to the identification of *maintenance and support* functions. For instance, there are specific performance expectations or measures associated with each block in an operational functional flow diagram. A check of the applicable functional requirement will indicate either a "go" or a "no-go" decision. A "go" decision leads to a check of the next operational function. A "no-go" indication (constituting a symptom of failure) provides a starting point for the development of a detailed maintenance functional flow diagram. The transition from an operational function to a maintenance function is illustrated in Figure 2.14. Figure 2.15 presents a more in-depth functional flow diagram.[11]

[11]It should be noted that all of the *forward* and *reverse* flow activities shown in Figure 1.20 should be covered through either the "operational" or the "maintenance and support" functional flow block diagrams (FFBDs).

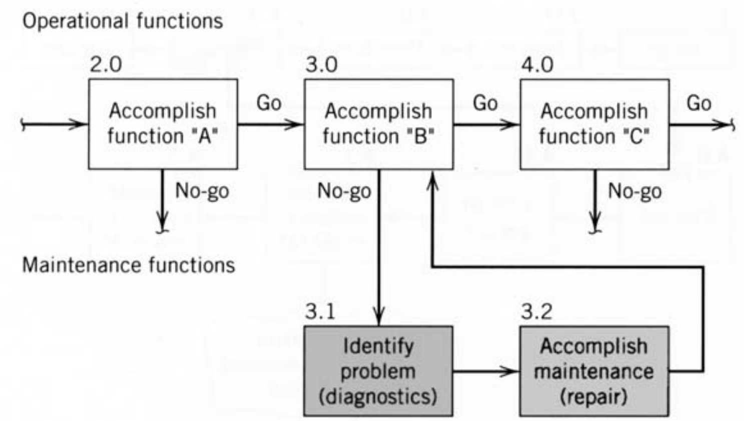


Figure 2.14 Transition from operational functions to maintenance functions.

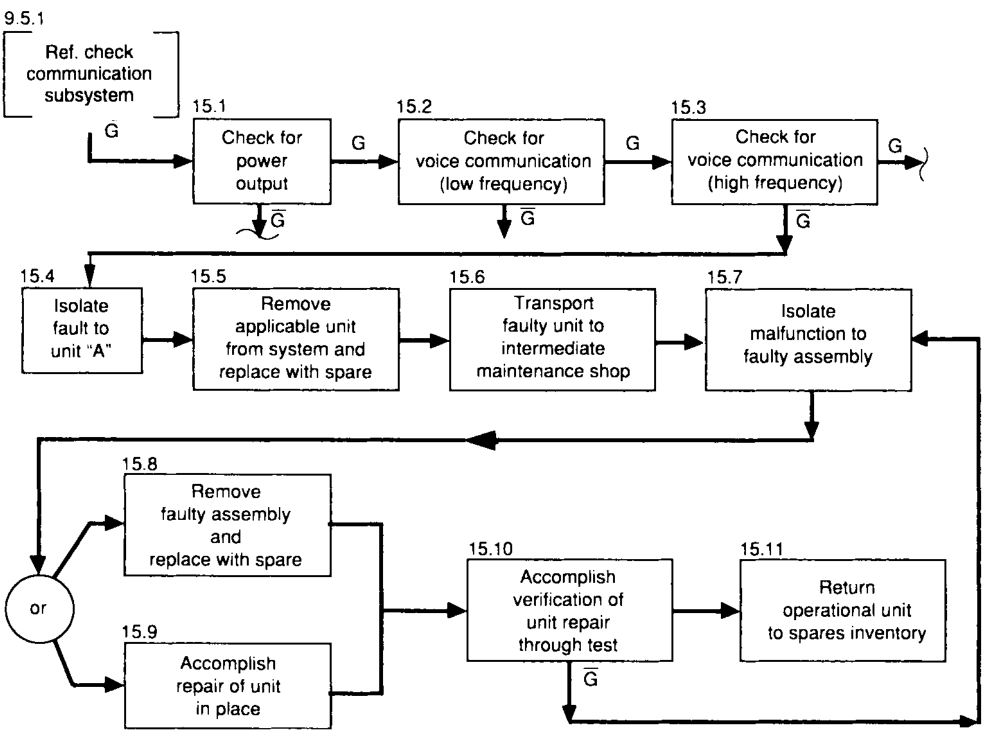


Figure 2.15 Maintenance functional flow diagram.

2.7.4 Application of Functional Analysis

The functional analysis provides an initial description of the system and, as such, its applications are extensive. Figure 2.16 illustrates a top-level operational functional flow diagram for a manufacturing system, commencing with the identification of need (block 1.0) and extending through system retirement (block 7.0). In areas where a greater degree of definition is desirable, the applicable block(s) may be broken down to a second level, third level, and so on, in order to gain the appropriate level of visibility necessary for the determination of resource requirements. In this instance, the ultimate manufacturing "operating" functions have been identified in the breakout of block 5.1.

For each of the blocks in Figure 2.16, the analyst should be able to specify *input* requirements, expected *outputs,* external *controls* and/or *constraints,* and the *mechanisms* (or resources) necessary to accomplish the specific function in question. In the process of identifying the appropriate resource requirements, there may be a number of alternative approaches that should be considered. Trade-off studies are conducted, alternatives are evaluated against criteria developed from the established technical performance measures (i.e., the TPMs derived in Section 2.6), and a preferred approach is recommended. It is at this point that one begins to identify the requirements for hardware, software, people, facilities, data, or combinations thereof. Figure 2.17 reflects the process that should be applied to each of the blocks in Figure 2.16.

In the evaluation of each functional requirement, the alternatives may include the selection of "commercial off-the-shelf" (COTS) items readily available from a number of different sources of supply, COTS items that may require some degree of modification, and/or "developmental" items that are unique to a particular application or where some new design is required. Past experience has indicated that extensive time and cost savings can be realized through the selection of readily available COTS equipment or reusable software, the utilization of existing facilities, and so on. Figure 2.18 illustrates the various options in this area.[12]

Figure 2.18 shows that it is essential that a *good* definition of the inputs and outputs (and the applicable metrics) be established if one is to fully understand not only the *interfaces* between the different functions identified in Figure 2.16, but the precise requirements in the process of resource identification. If these input-output requirements are not well defined, the decision-making process as to a preferred approach becomes difficult; thus leading to the possibility of initiating a new costly design and development effort when, in actuality, an existing off-the-shelf item could fulfill the need.

[12]In recent years there has been considerable emphasis on the utilization of commercial off-the-shelf (COTS) items, versus the pursuit of new design efforts at the detailed component level, in the development of new systems and in the modification and upgrade of existing systems that are currently in the inventory. This has been particularly applicable in the acquisition and upgrade of defense systems. The objectives are to reduce the time involved in the development and acquisition of new systems, improve system supportability/serviceability through the utilization of standard components that can be easily backed up with readily available spares and repair parts obtainable from multiple sources of supply, and to reduce costs from a life-cycle perspective.

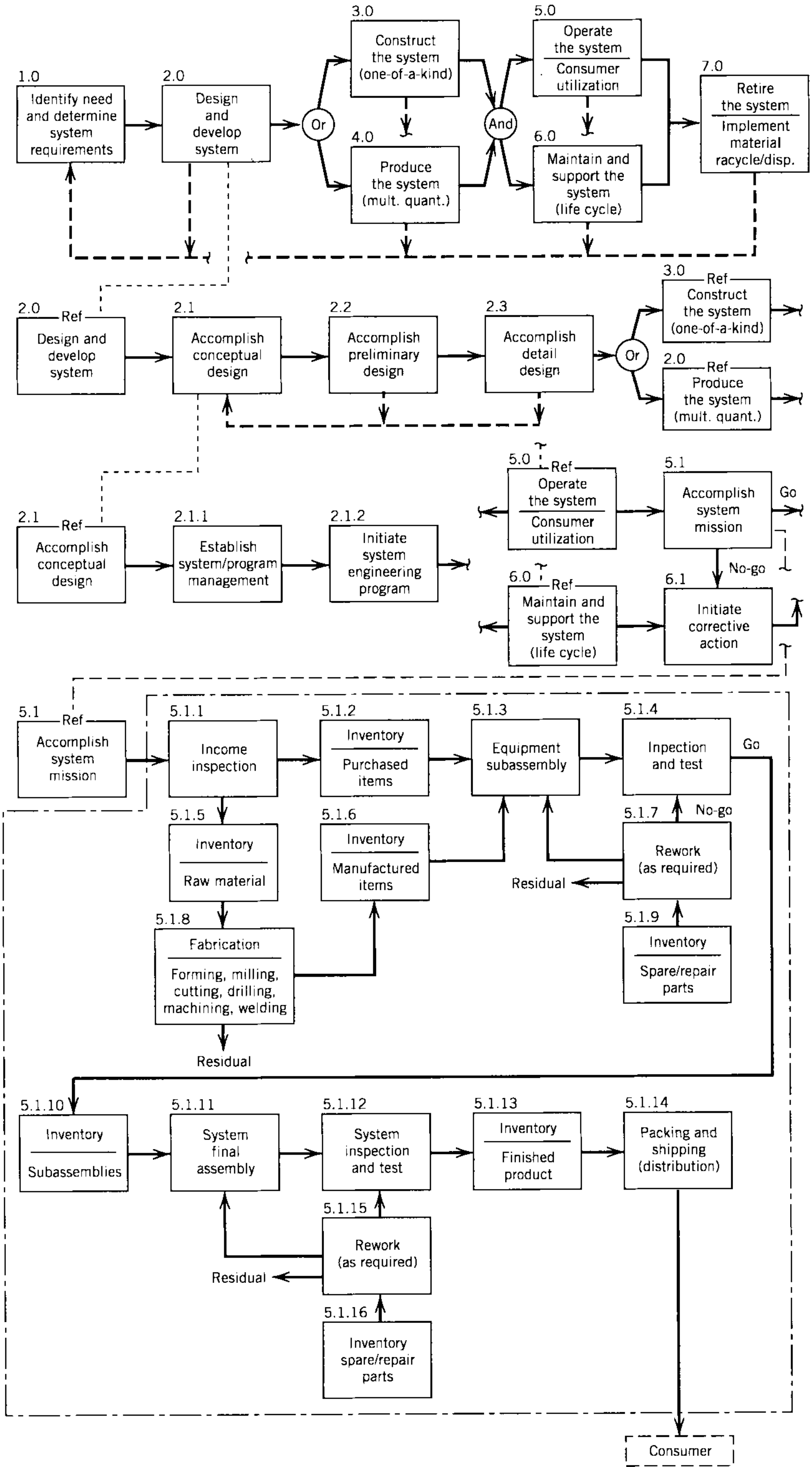


Figure 2.16 Functional flow diagram for a manufacturing system.

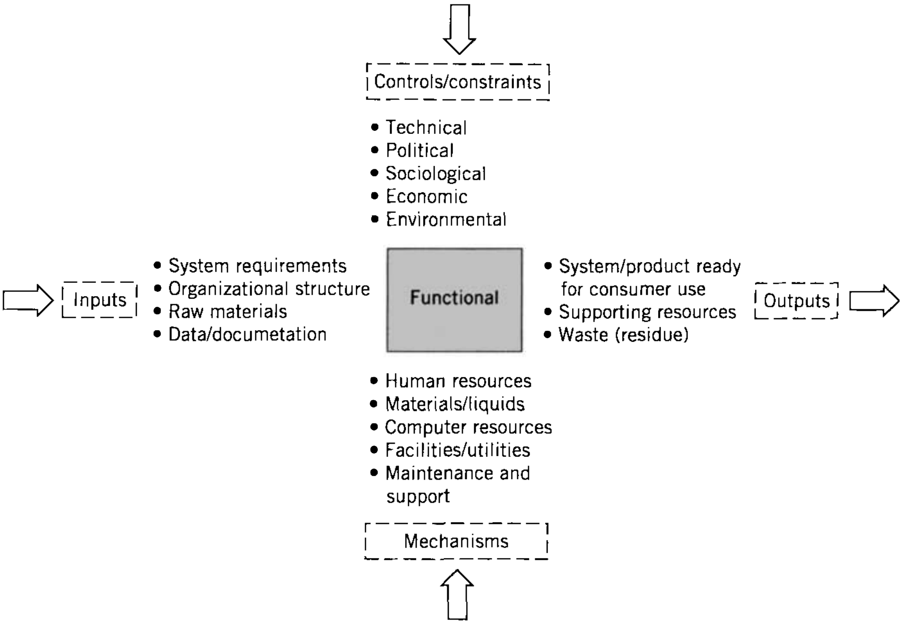


Figure 2.17 Identification of resource requirements (i.e., "mechanisms").

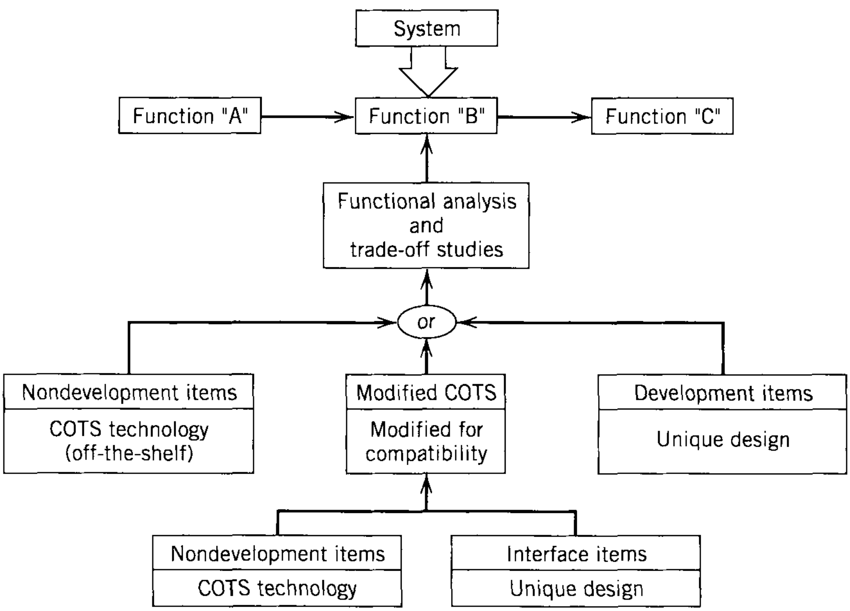


Figure 2.18 Identification of commercial off-the-shelf (COTS) items from functional analysis.

The functional analysis can facilitate an "open-architecture" approach to system design. A good comprehensive functional description of the system, with the interfaces well defined (both qualitatively and quantitatively), can lead to a structure that will not only allow for the rapid identification of resource requirements, but permit the possible incorporation of new technologies later. The objective is to design and develop a system that can be easily modified, through the insertion of new technologies, without causing a "costly" redesign of all of the elements of the system in the process.

In many current situations, the requirements in design are changing from a detailed "design to the component level" to the design of systems using a "black-box-integration" approach. Given the need to reduce acquisition times, while responding to an ever-changing set of requirements on a continuing basis and with many more suppliers involved, the system *architecture* must allow for the ease of upgrade and/or modification. In other words, the system *structure* must be such as to facilitate design on an *evolutionary* basis, and with minimum cost. This can be enhanced through a good and comprehensive functional definition of the system in the early conceptual design phase of the life cycle.

Figure 2.19 illustrates a manufacturing system in which there are many suppliers (on various locations throughout the world) who produce components for a consumer product that must be effectively integrated and tested. There are fabrication functions, subassembly functions, assembly functions, and test functions. Where, in many instances in the past, the manufacturing activity involved a bottom-up "build" approach, the challenges today relate to the *integration* of the various components into the end

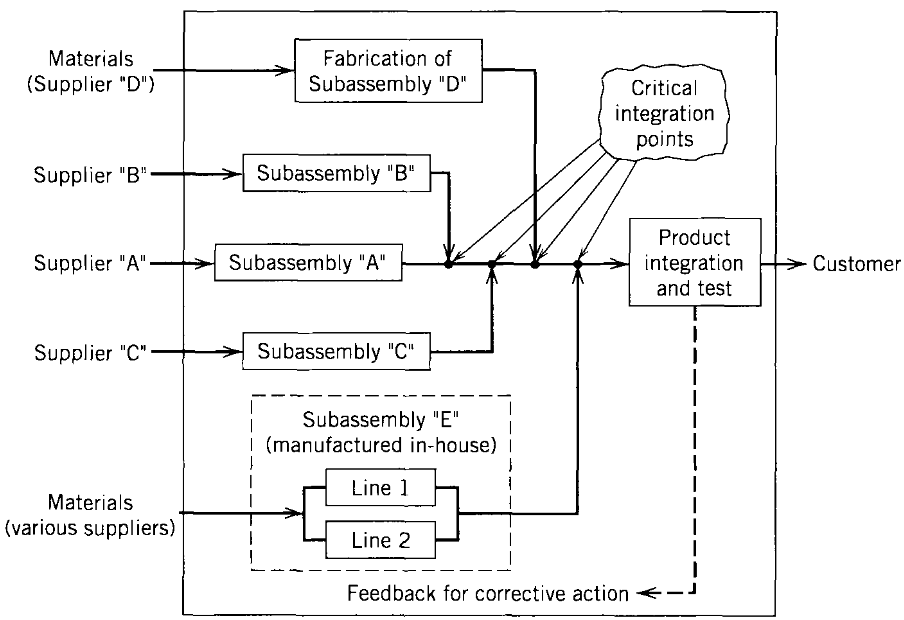


Figure 2.19 Manufacturing system (critical integration points).

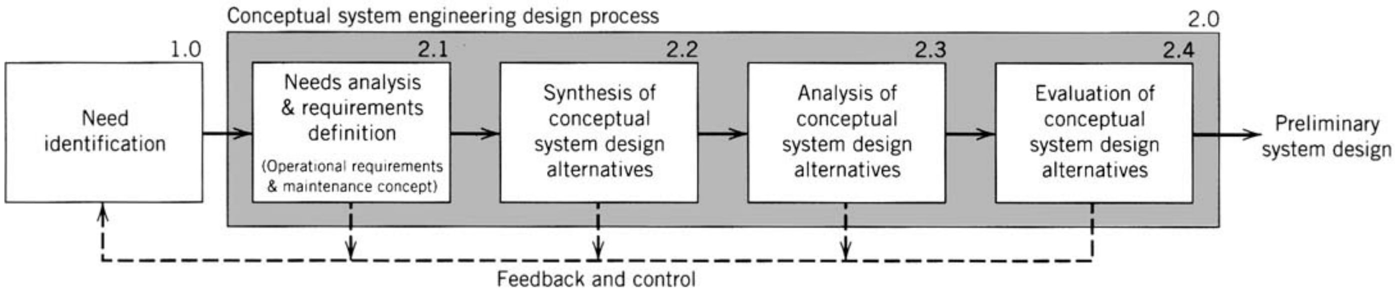
product. Without a good early definition and specification of the functional interfaces, the final integration and test activity may result in a costly trial-and-error process. In Figure 2.19, the example reflects a factory where the subprocesses were being accomplished effectively and efficiently; however, there were considerable problems associated with the "integration" activities—that is, the four critical integration points. The functional interfaces were not well defined from the beginning, causing a great deal of modification and rework downstream.

In completing a functional analysis, care should be taken to ensure that the required resources are properly identified for each function. A time line analysis may be performed to determine whether the functions are to be accomplished in series or in parallel. It may be possible to share resources in some instances; that is, the same resources may be utilized to accomplish more than one function. The identified resources may be combined and integrated to the extent possible. Every effort should be made to avoid the specification of resources that are not necessary. Figure 2.20 illustrates a documentation format that can be applied to formalize the identification of such resources.

In summary, a functional analysis is a critical step in the early system design and development effort, and it forms a baseline for many activities that are conducted subsequently. For instance, it serves as a basis in the development of the following:

1. Electrical and mechanical design for functional packaging, condition monitoring, and diagnostic provisions
2. Reliability models and block diagrams
3. Failure mode, effect, and criticality analysis (FMECA)
4. Fault-tree analysis (FTA)
5. Reliability-centered maintenance (RCM) analysis
6. Maintainability analysis
7. Human factors analysis
8. Operator task analysis (OTA)
9. Operational sequence diagrams (OSDs)
10. System safety/hazard analysis
11. Security analysis
12. Level of repair analysis (LORA)
13. Maintenance task analysis (MTA)
14. Logistics analysis (supply chain analysis)
15. Supportability/serviceability analysis
16. Operating and maintenance procedures
17. Producibility analysis
18. Disposability and material recycling analysis

In the past, the functional analysis has not always been completed in a timely manner, if completed at all. As a result, the various design disciplines assigned to a given program have had to generate their own analyses in order to comply with program



Activity number	Activity description	Required inputs	Expected outputs	*Resource requirements* (activities/techniques)
1.0	Need identification	Customer surveys; marketing inputs; shipping and servicing department logs; market niche studies; *competitive product research*	A specific qualitative and quantitative needs statement responding to a current deficiency. *Care must be taken to state this need in* functional terms.	Benchmarking; statistical analyses of data (i.e., data collected as a result of surveys and consolidated from shipping and servicing logs, etc.)
2.1	Needs analysis and requirements definition	A specific qualitative and quantitaive needs statement expressed in functional terms.	Qualitative and quantitative factors pertaining to system performance levels, geographical distribution of products, expected utilization profiles, user/consumer environment; operational life cycle, effectiveness requirements, the levels of maintenance and support, consideration of the applicable *elements of logistic support*, the support environment, and so on.	Quality Function Deployment (QFD); input-output matrix; checklists; value engineering; statistical data analysis; trend analysis; matrix analysis; parametric analysis; various categories of analytical models and tools for simulation studies, trade-offs, etc.
2.2	Synthesis of conceptual system design alternatives	Results from needs analysis and requirements definition process; technology research studies; supplier information	Identification and description of candidate conceptual system design alternatives and technology applications.	Pugh's concept generation approach; brainstorming; analogy; checklists.
2.3	Analysis of conceptual system design alternatives	Candidate conceptual solutions and technologies; results from the needs analysis and requirements definition process	Approximation of the "goodness" of each feasible conceptual solution relative to the pertinent parameters, both direct and indirect. This goodness may be expressed as a numeric rating, *probabilistic measure, or fuzzy* measure.	Indirect system experimentation (e.g., mathematical modeling and simulation); parametric analyses; risk analyses.
2.4	Evaluationof conceptual system design alternatives	Results from the analysis task in the form of a set of feasible conceptual system design *alternatives.*	A specific qualitative and quantitative needs statement responding to a current deficiency. Care must be taken to state this need in functional terms.	Design-dependent parameter approach; generation of hybrid numbers to represent candidate solution "goodness"; conceptual system design evaluation display.

Figure 2.20 Document format for resource requirements.

requirements. In many instances, these efforts were accomplished independently, and many design decisions were made without the benefit of a *common* baseline to follow. This, of course, resulted in design discrepancies and costly modifications later in the system life cycle.

The functional analysis provides an excellent and very necessary baseline, and all applicable design activities must "track" the same data source in order to meet the objectives for system engineering, as stated in Chapter 1. For this reason, the functional analysis is considered a key activity in the system engineering process.

2.8 REQUIREMENTS DEFINITION AND ALLOCATION

Having defined the basic architecture for the system, the discussion continues with the "requirements analysis" process and defines the specific *input* design criteria for the various subsystems and lower-level components of the overall system. With Figure 1.14 (blocks 0.1/0.2) defining the top system-level requirements, it is now essential to define the specific "design-to" requirements (criteria to which the system should be defined) for critical items of equipment, major software modules, applicable facilities, personnel, elements of support, and so on, that have been identified through the functional analysis. The requirements for the system must be allocated (or apportioned) down to its various components as appropriate. Conversely, the composite of these requirements for the various components, when combined, must support the initially specified requirements for the overall system. Basically, this is a top-down requirements distribution process, which is somewhat iterative initially and often resulting from trade-offs conducted horizontally across the spectrum of system components. The ultimate objective is, of course, to be able to define specific qualitative and quantitative *design requirements* for each significant element of the system and to include such requirements in the appropriate specification for use in the procurement and acquisition process.[13]

2.8.1 Functional Packaging and Partitioning

Given a top-level description of the system, the next step is to break the system down into its components by *partitioning*.[14] This involves a breakdown of the system into subsystems and lower-level elements such as illustrated in Figure 2.21. Such elements

[13]As conveyed in Chapter 1 (Figure 1.1), there is a great deal of "outsourcing" taking place, and there appears to be considerable growth in the number of "suppliers" involved in a typical large-scale project. Whenever a new supplier is selected, a "specification" is prepared, which constitutes a critical part of the data package for the purposes of procurement and subcontracting. Because many of the major elements of a system are now being subcontracted, it is essential that a complete and well-defined set of *requirements* be included in each applicable specification. Further, there must be a top-down/bottom-up *traceability* of requirements throughout the hierarchy of specifications for a specific system. Specifications are discussed further in subsequent chapters.

[14]The concepts of system *architecture* and *partitioning* are discussed further in E. Rechtin and M. Maier, *The Art of Systems Architecting* (Boca Raton, FL: CRC Press, 1996).

Next Page

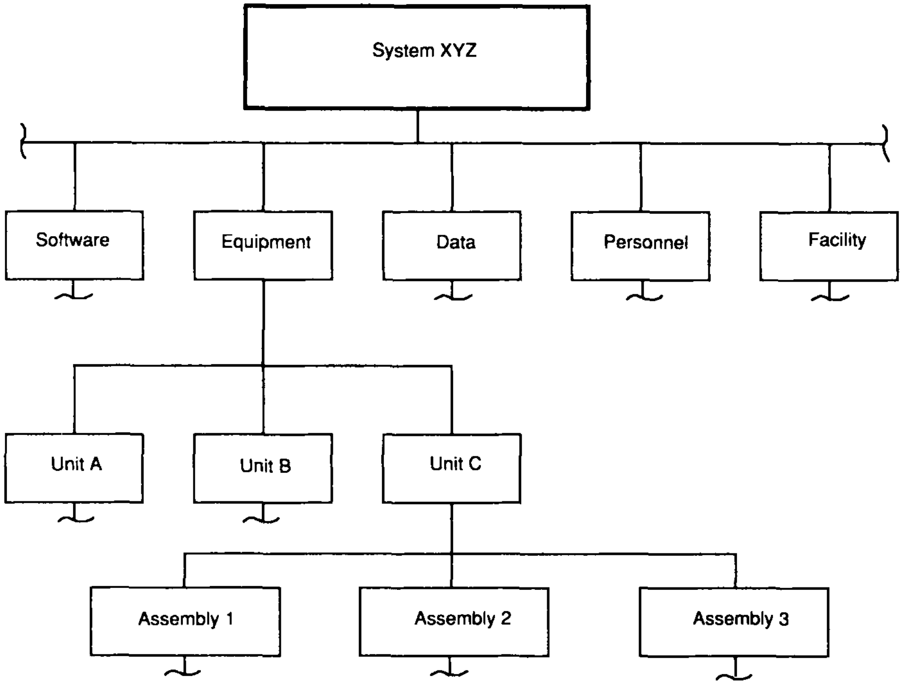


Figure 2.21 Hierarchy of system components.

are initially identified through the functional analysis and evaluation of each function on an individual-by-individual basis (see Figure 2.17). The challenge subsequently is to identify and group closely related functions into packages, employing a common resource (e.g., equipment, software, facilities) to accomplish multiple purposes to the extent possible. Although it may be relatively easy to identify individual functional requirements and associated resources on an independent basis, the results may be rather costly when it comes to system packaging, weight, size, and so on. The basis questions are, What hardware or software can be selected that will perform multiple functions? How can new functional capabilities be added in the future without adding any new physical elements to the system structure (i.e., growth potential)? Can any physical resources (e.g., equipment, software, facilities) be deleted without losing any of the required functional capabilities previously defined?

The partitioning of a system into its elements is evolutionary in nature. Common functions may be grouped or combined in such as way as to provide a system packaging scheme, to meet the following objectives:

1. System elements may be grouped by geographical location, by nationality, by a common environment, or by similar types of equipment and/or software.

2. Individual system "packages" should be as independent as possible with a minimum of "interaction effects" in relation to other packages. A design objective is to