4.5 Synthesis

Synthesis is the creative process that translates requirements (performance, function, and interface) into alternative solutions resulting in a physical architecture for the "best value" design solution, consisting of people, products, and process solutions for the logical, functional grouping of the requirements. In the Synthesis process, design engineers first conceive and then later refine specific designs that will serve to satisfy operational needs.

The Synthesis process defines design solutions and identifies systems that will satisfy the program requirements. Synthesis translates the requirements, as set in context by the functional architecture, into the design architecture, consisting of the physical architecture with its associated technical requirements. The resulting architecture provides an arrangement of system elements by designing their composition and interfaces, both internal and external. Additionally, the design architecture incorporates environmental, technical, and other constraints.

Synthesis is seldom, if ever, a one-step process, but rather accomplished many times over the life of a project in response to many factors. These include newly evolving technology, test data from the present or previous designs, changes in requirements from the user, changes in the price or availability of components, and feedback from the field once a system is deployed. As with all System Engineering (SE) functions, different objectives and activities exist within different phases of the acquisition process.

4.5.1 Introduction

The Synthesis process is an element of the overall SE discipline, with other processes occurring before, during, and after. Synthesis also leverages the efforts conducted under various Specialty Engineering (Section 4.8) disciplines through concurrent engineering. Accordingly, Synthesis requires a number of inputs into the process in order to achieve the anticipated results, or outputs, of the process. See Figure 4.5-1.

Synthesis is conducted to translate the requirements (based on the functional architecture) into a physical architecture by defining and allocating the system elements. Those elements are then refined and integrated into the system's physical configuration, which satisfies the functional and performance requirements. This process relies heavily on prior establishment of clearly defined, documented, and validated requirements.

When entering the Synthesis process, do not assume that the entire requirements set associated with the functional area under consideration is achievable within the cost and schedule constraints. However, do assume that all requirements associated with the functional area under consideration have been validated in accordance with Validation and Verification (Section 4.12). The engineers involved in Synthesis work to find the best possible solution that will optimize achievement of the program requirements for the functional area under consideration. This requires close and continual coordination with Requirements Management (Section 4.3) and Functional Analysis (Section 4.4).

Success of the Synthesis or design process relies on a structured and disciplined approach to achieving the desired outcomes. The Synthesis outputs will naturally emerge from taking the appropriate steps during the design process. Conducted properly, Synthesis defines the build-to characteristics of the system or system elements. The Configuration Items (CI) are established and defined during Synthesis. At each level of the resulting design architecture, the requirements and interfaces must be verified. The Synthesis process must not only identify technically feasible and programmatically achievable design alternatives, but the alternatives must also be well analyzed, documented, and finally placed under disciplined management.

Process:

Perform Synthesis

ID No.: 4.5 (iCMM PA03 & 04)
Date: March 25, 2000
Revision Date: August 30, 2006

Next Higher Level Process:
Perform System Engineering

Process Owner:
System Engineering Council

Process Objective:

The creative process that translates requirements (performance, function, and interface) into alternative solutions resulting in a physical architecture for the “best-value” design solution, consisting of people, products, and process solutions for the logical, functional grouping of the requirements.

Inputs

a) Constraints, Legacy System, Market Research, Standards, Technology
b) SEMP, WBS
c) Requirements, RVCD
d) Functional Architecture, OSED
e) Trade Study Reports
f) IRDs, ICDs
g) DARs
h) Analysis Criteria
i) Risk Mitigation Plans, Constraints
j) Configuration Status Accounting Report ,Baselines, Baseline Changes,
k) Requirements
l) Life Cost Estimate, Constraints
m) SE Processes, SE Best-Practices Documentation (SEM), SEBOK

Providers

a) EXT
b) ITP
c) RM
d) FA
e) TS
f) IM
g) SpecEng
h) IA
i) RSK
j) CM
k) V&V
l) LCE
m) MSE

PROCESS TASKS

Beginning Boundary Task
Requirements analysis

- Review requirements and define objectives
- Identify potential alternative
- Define solution set
- Allocation of requirements and design constraints to system elements
- Define design and performance characteristics
- Define physical architecture
- Analyze and define alterinatives
- Assess requirements compliance
- Select “Best Value” Alternative

Ending Boundary Task
Document alternatives

Lifecycle Phase

- ☑ Mission Analysis
- ☑ Investment Analysis
- ☑ Solution Implementation
- ☑ In-Service Management
- ☐ Disposal

Outputs

a) Physical Architecture
b) Description of Alternatives
c) Product Definition
  - -Configuration item description
  - -Specificaion inputs
  - -Requirements compliance matrix
  - -Work breakdown structure
d) Constraints
  - – Design constraints
  - – Trade study requests
e) Planning Criteria
f) Tools/Analysis Requirements
g) Concerns/Issues
h) Operational Prototype Results

Customers

a) RM, FA, TS, IM, SpecEng, V&V, EXT, LCE, ITP
b) TS, SpecEng
c) CM, RM
d) TS, LCE, RM
e) ITP
f) IA
g) RSK
h) IM, V&V, LCE

Figure 4.5-1. The Synthesis Process-Based Management Chart

4.5.2 Process Inputs

The Synthesis process starts at the conclusion of preceding key SE steps, as illustrated in Figure 4.5-2. These SE processes result in a number of outputs that will serve as necessary inputs to Synthesis.

Like Synthesis, the processes preceding it are not necessarily one-step processes. Each may undergo a number of iterations through the given process before the output is ready for the next process to begin. Additionally, the Requirements Management (Section 4.3) and Functional Analysis (Section 4.4) processes are tightly coupled, and a few iterations through these processes will occur before the outputs are ready to proceed into Synthesis.

Once it begins, Synthesis will be an iterative process, at times looping back through Requirements Management. This is known as the requirements verification loop. Synthesis might also at times initiate iteration back through Functional Analysis, known as the design loop. During these iterative loops through preceding processes, the program requirements and/or

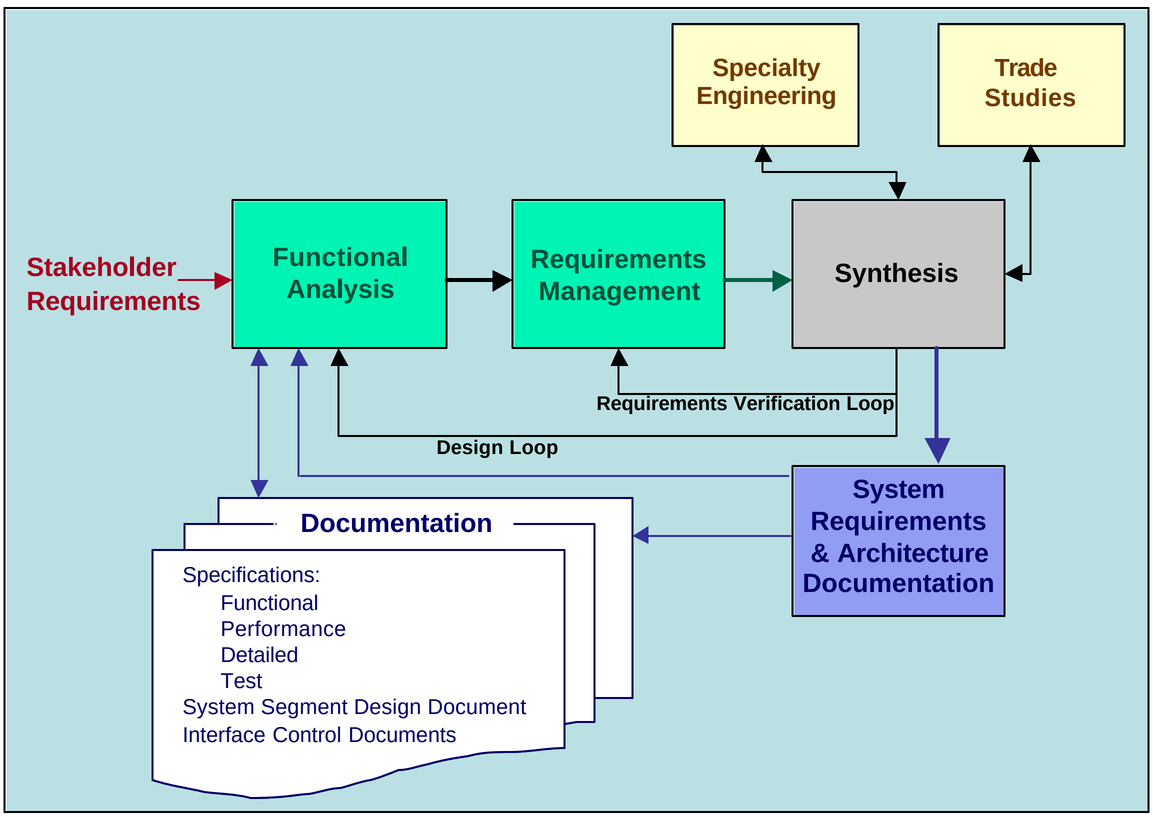

Figure 4.5-2: Requirements and Architecture Definition

functional architecture are constrained and refined to optimize the potential for viable design alternatives. This ensures that the functional architecture and requirements at lower levels of the physical architecture reflect the envisioned design.

4.5.2.1 Initial Inputs

The inputs resulting from the previously conducted SE processes are known as the initial inputs because they serve to initiate Synthesis. They must be available before the start of system design.

4.5.2.1.1 Functional Architecture

During Functional Analysis (Section 4.4), the high-level functions are decomposed to lower level functional groups or areas that can be satisfied by system design alternatives. The functional architecture must describe the functional arrangements and sequencing of subfunctions resulting from decomposition. The functional architecture does not consider design solutions, but only tasks or functions that the solution(s) must perform. Synthesis, by contrast, considers the grouped and decomposed functions, or functional areas, in light of technically feasible and achievable solutions.

Functional Analysis provides the design group the appropriate area of the functional architecture at which to begin the design process. This functional architecture is translated into an established requirements set that documents the problem or set of problems to be solved by Synthesis. The problem for the design group is to identify and define a system or systems that will adhere to the prescribed functional architecture while meeting stakeholder requirements.

4.5.2.1.2 Program Requirements

The user needs and system functions are translated into a set of clearly defined, prioritized, measurable, and validated requirements (Section 4.3) for which the design group must provide a solution or solution set. The established program requirements (either preliminary Program Requirements (pPR) or final Program Requirements (fPR)), documented in the Exhibit 300 Attachment 1, dictate the tasks the system(s) under design must perform through functional requirements. The program requirements dictate how well the system(s) must perform its tasks through documented performance requirements. And finally, the program requirements ensure system compliance, function, and performance through measurable verification requirements on the Requirements Verification Compliance Document (RVCD).

Not only will information be needed regarding what the system must perform, how well it will be performed, and how performance will be measured, but the program requirements also establish the system's limitations. The program requirements contain the constraint requirements levied on potential solutions. Design constraints further limit the system under design from reaching its desired level of achievement. System design usually faces limitations; therefore, design constraints must be identified, documented, and managed so that they do not manage design by default. Acknowledged or not, the constraints determine the output of the system under design.

During the Synthesis process, the design engineers must consider the limitations of engineering. Often, "the laws of physics" or the "state of the art" limits solutions. The design engineers need to clearly understand technical as well as programmatic limitations to trade risk, schedule, and financial constraints in overcoming challenges to satisfying the program requirements.

4.5.2.1.3 Legacy System Definitions

In the FAA, it is rare when a solution is introduced into a pristine environment (i.e., an environment where a system is not already satisfying user needs.) It is also rare that established needs do not evolve and change as the operational environment evolves and changes. Consequently, it is important to understand the existing legacy system that currently seeks to satisfy documented needs.

Understanding must include knowledge of the legacy system functions, performance, and its shortfalls. Only then can the design solution provide an alternative that improves existing capabilities, adds new functionality, and complies with evolving user needs. All documentation regarding system functional, performance, and constraint requirements is therefore a necessary input into the Synthesis process.

The design constraints imposed by the need for the system to operate with existing interfacing systems must also be understood. Interface Control Documents (ICD) will provide the information to ensure integration into the existing environment.

Finally, the new system must eventually operate in the existing support environment. Documentation regarding legacy system maintenance and support is needed to ensure that the system is designed in a manner that will enable it to continue to perform the needed user tasks at the needed level of performance once introduced into the support system.

4.5.2.1.4 Implementation Strategy and Planning

The Implementation Strategy and Planning (ISAP), the Exhibit 300 Attachment 3, is the document within the Acquisition Management System (AMS) that provides the strategy and planning for the detailed actions and activities necessary to execute the program within the cost, schedule, and performance constraints. The ISAP encompasses all elements of program implementation. This may include the acquisition of systems and equipment, construction or modification of facilities and the physical infrastructure, functional integration of planned capabilities within the existing infrastructure, and procurement of services.

To perform Synthesis, one must also know the schedule or budget constraints. If an ISAP exists, it provides this needed information. If such a plan does not exist, the design team will have to determine the cost and schedule constraints through interface with program management and other stakeholders.

4.5.2.1.5 Operational Services and Environment Description (OSED)

The OSED provides operational, safety, performance, and interoperability requirements. (See Functional Analysis (Section 4.4.5.4).) This document provides needed information for the Synthesis process. The OSED identifies the desired air traffic services and/or capabilities and their operational environments, including documented operational functions, performance expectations, and selected technologies. It defines the customer needs so that more appropriate alternative selections are considered during Synthesis.

4.5.2.1.6 Preliminary Work Breakdown Structure (WBS)

A preliminary WBS is provided and initially guides Synthesis efforts. (See Integrated Technical Planning (Section 4.2).) It is then refined under Synthesis by incorporating the characteristics necessary to support the functional and selected physical architecture(s) of potential design alternatives. The WBS defines categories of work, work packages, and, ultimately, through Synthesis, identifies associated physical elements. The WBS is invaluable from the planning and management perspective, since it establishes a top-down framework for allocating and computing costs. The WBS assists in tracking the status of engineering efforts, resource allocations, cost estimates, expenditures, and cost and technical performance.

During Synthesis, the WBS must be scrupulously maintained and finalized to show in a hierarchical manner all work elements needed to complete a given program or project. As solution physical architectures are defined, the physical elements are introduced into the WBS.

4.5.2.2 Other Inputs

Beyond the inputs available from SE processes occurring prior to Synthesis, there will be inputs gathered during Synthesis from sources both internal and external to the SE process.

4.5.2.2.1 Market Research

Market research is conducted during Synthesis to gather data to conduct the process as well as for various other reasons. During the phases of the AMS cycle, the role of market research in the Synthesis process will vary.

The first time through the Synthesis loop, when a pPR database has been established and provided as input to the Synthesis process, market research helps determine the available technologies or various systems that can meet all or part of the program requirements.

If multiple viable alternatives do not exist, the program requirements and functional architecture will be modified for optimization of alternative solutions. This optimization can occur numerous times as needed. During the final Synthesis iteration, the fPR is approved, and market research is conducted in concert with the design team to identify vendors that meet the finalized program requirements.

One final and important consideration for market research is to determine the market base for proposed design alternatives. A smaller potential market base for a system and/or its components will inevitably translate to an increase in cost risk and a greater potential for the market not to continue to produce the needed items for the needed timeframes as the demand for the supply diminishes. Market research is therefore valuable in determining not only what is available in the marketplace, but also in determining the extent of its availability and the likelihood that it will continue to be available for the required project/program lifecycle for which Synthesis will provide a solution.

4.5.2.2.2 Risk Mitigation Plans

Risk Mitigation Plans, although invaluable, may or may not be available for a given iteration through the Synthesis loop. For the initial time through the Synthesis loop, the fPR and functional architecture are not available. Therefore, the risks associated with potential design alternatives are undefined, and concerns and issues associated with those risks are not yet forwarded to the Risk Management process (Section 4.10) by the Synthesis team.

Subsequent iterations through the Synthesis loops, however, will have incorporated those initial concerns and issues, and a risk mitigation plan will have been developed under the Risk Management process (in concert with the Synthesis process).

4.5.2.2.3 Trade Study Reports

Trade Study reports are invaluable, whether available to the Synthesis process from previous related efforts or whether solicited through the course of the process. The Trade Study report provides documented answers to many issues and concerns for the Synthesis process, such as the feasibility of design alternative, the state of technology to support the alternative, and so on.

Existing Trade Study reports should identify related technologies that Synthesis may consider for incorporation into design alternatives. These reports provide valuable insight into what is feasible given the current state of the art.

When the Trade Study is conducted in concert with Synthesis, it is geared toward exploring and determining feasibility, associated risks, maturity of design, conformance to the program requirements and functional architecture, and adherence to the various constraints to the program/project. This input is solicited in the sense that the Synthesis process works in concert with the Trade Study process to determine objectives and needed outcomes for the Trade Study report. (See Trade Studies (Section 4.6).)

4.5.2.3 Summary of Needed Input for Synthesis

Availability of data depends on the status of the Synthesis process. If it is the first-time entry into Synthesis, or the first Synthesis loop, not all data will be available. However, as the Synthesis process continues, more data becomes available from other SE disciplines. Table 4.5-1 summarizes the data that is required and its availability for the Synthesis process.

Table 4.5-1. Needed Synthesis Data

Input	Delivering Process	SEM Reference	Availability
preliminary Program Requirements	Requirements Management	Section 4.3	First and subsequent loops
Functional Architecture	Functional Analysis	Section 4.4	First and subsequent loops
Legacy System Specifications	External to SE	N/A	First and subsequent loops
Legacy Interface Requirements	Identify, Define and Control Interfaces	Section 4.7	First and subsequent loops
Draft ISAP	Integrated Technical Planning	Section 4.2	First Synthesis loop
Operational Services and Environment Description	Functional Analysis	Section 4.4	First and subsequent loops
Preliminary WBS	Integrated Technical Planning	Section 4.2	First Synthesis loop
Market Research	External to SE	N/A	May not be available first loop through Synthesis
Trade Study Report	Trade Studies	Section 4.6	May not be available first loop through Synthesis
Risk Mitigation Plans	Risk Management	Section 4.10	May not be available first loop through Synthesis

4.5.3 Process Steps

Synthesis activities involve selecting a preferred solution or arrangement from a set of alternatives and understanding associated cost, schedule, performance, and risk implications. Synthesis entails undertaking a number of distinct steps to achieve measurable goals and objectives while striving to manage or overcome constraints. Alternative candidate designs are first conceptualized, and then candidate alternative solutions are defined and refined to meet the established program requirements.

Engineering analysis is used, as necessary, to evaluate alternatives. Evaluation will identify, assess, and quantify risks and select proper risk mitigation approaches. The risk management plan, if available, is used to refine the various design alternatives and achieve a balance between risk and technical progress. Too much risk within a given alternative could result in an unachievable design at the end. Assuming too little risk within a given alternative could also result in a solution that cannot be reached within the schedule constraints established for the project. These two extremes are balanced against the program requirements and established functional architecture through the guidance provided in the Risk Mitigation Plan(s). (See Integrated Technical Planning (Section 4.2)).

Analysis of alternative solutions also results in an understanding of cost, schedule, and performance impacts. As subsystem requirements are defined, identification of the needs, requirements, and constraints for lifecycle processes is completed. Figure 4.5-3 identifies the specific tasks that define Synthesis.

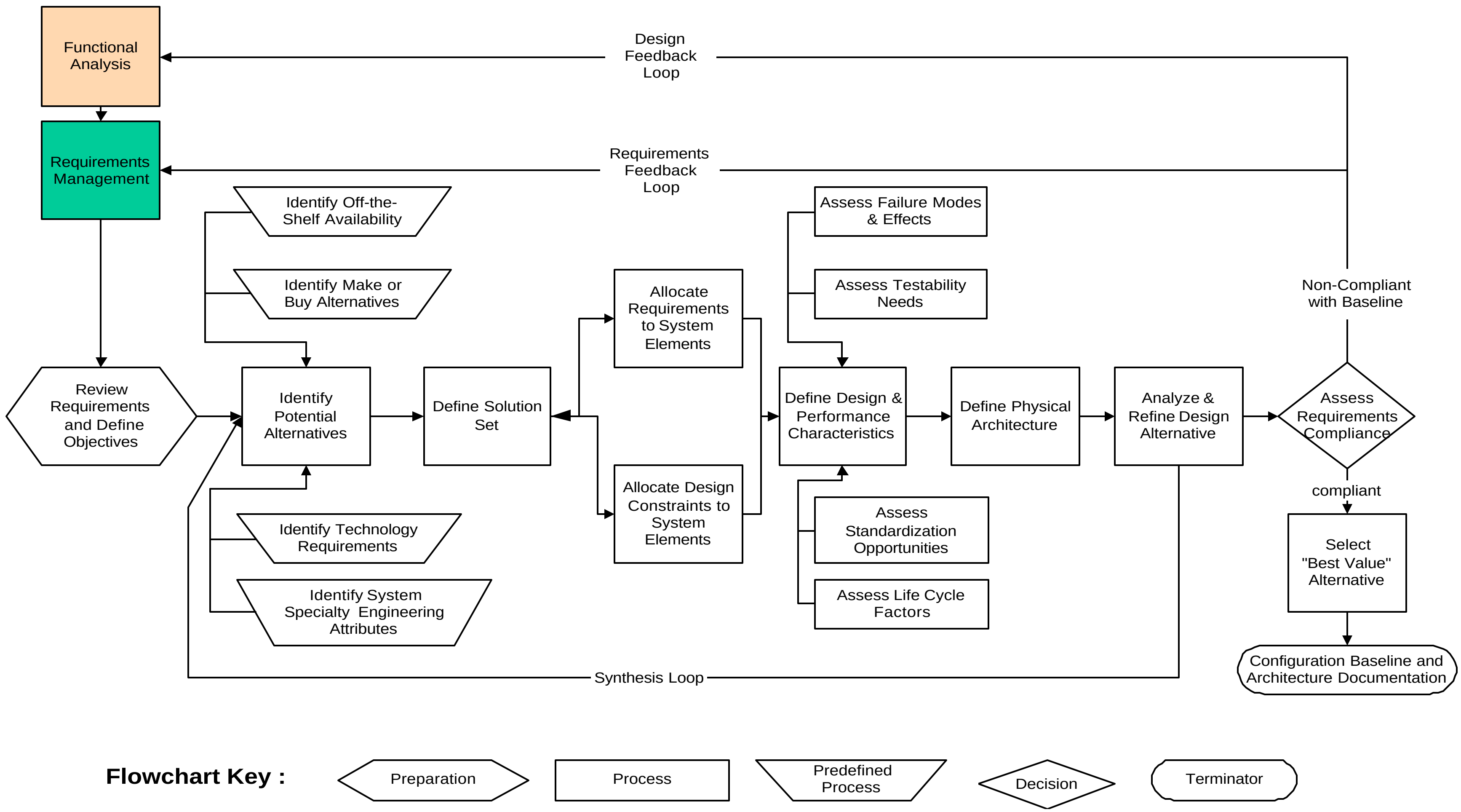

Figure 4.5-3: The Synthesis Process Activities

Synthesis demands creativity to achieve success. The ability to discover new solutions, to examine the requirements from new perspectives, and to formulate new concepts from two or more previously held ideas challenges the design group during this process. For the design team to succeed, each member must exercise awareness and sensitivity to problems associated with each proposed approach. Each person must exercise flexibility, originality, self-discipline, and persistence while maintaining adaptability, nonconformity, tolerance for ambiguity, self-confidence, and a healthy skepticism.

In addition to exercising individual characteristics, the team must also be aware of group characteristics and dynamics that are essential for successfully developing achievable yet satisfactory design alternatives.

A group of "*like-thinkers*" typically arrives at a mutually agreeable solution, or solution set, in less time and with less discourse than a diverse group with differing perspectives and priorities. The solution reached in this relatively pain-free manner will not have always considered and analyzed every facet of the approach and all problems associated with it. As a result, the solution may not in the end satisfy all the requirements and design constraints levied on the Synthesis process. The devil's advocate plays an important role in the group and is as important to achieving the group's goals as the consummate politician.

Once a diverse and well-balanced group is formed, the group, through various methods, can begin to develop design alternatives and a set of prioritized objectives. The group can use such methods as brainstorming, brainwriting, and dynamic confrontation (see text box at right). Whatever method or combination of methods is selected for this creative development of alternatives, the group should take care to ensure that no individual is allowed to dominate the group and, therefore, its outcomes. Likewise, the group must ensure that every member of the group has ample opportunity to contribute to the group's efforts.

Brainstorming

This technique involves both idea generation and idea reduction. First, idea generation occurs by simply identifying as many solution ideas as possible. Later, in idea reduction, those potential solutions are ranked into groups, with a specific group encompassing those potential solutions considered most useful to the group.

This technique is frequently considered a powerful one, as it often results in the most creative and effective solutions. These solutions may arise from a combination of seemingly unrelated ideas generated early in the process. Brainstorming encourages creative and original thinking.

Brainwriting

This technique builds on the concept of brainstorming, as it is the same technique but simply replaces verbal communication with writing. Using this technique, team members will write down a number of relevant ideas on a sheet of paper (usually limited to three ideas). The paper is then passed to another team member who then develops those ideas. New ideas and elements are added to the original concept(s) and the augmented pages are then passed to another team member.

This process continues until each team member receives back the sheet of paper containing the original concepts he/she created. At this point, the beginning phase is complete, and a group leader collects all idea/solution sheets.

The next phase begins with all sheets being handed out to the entire group. The group then works to revise the ideas developed in the prior phase.

This technique alleviates one of the problems associated with brainstorming: it prevents dominant members from easily steering the efforts of the entire group.

Dynamic confrontation

This technique is an adversarial group process. The main idea is for team members to criticize every idea. A presentation is first made and then every element and assumption of that idea is intensely challenged. This technique tests every idea thoroughly and forces all members to thoroughly think through and develop their ideas.

4.5.3.1 Review Requirements and Define Objectives Definition—Step 1

After ensuring that all needed available Synthesis data has been gathered (see Table 4.5-1 above), Synthesis begins with a review of the program requirements and the functional architecture in order to understand what is to be performed and at what level of performance to meet stakeholder needs. Requirements Management will not dictate *how* the stakeholder needs will be met. The Synthesis process determines how to achieve stakeholder needs.

Establishing objectives assists in optimizing adherence to the requirements set within the technological and programmatic limits imposed on the design process. Objectives must be linked to stakeholder needs and system requirements. Objectives take into consideration, but are not limited to, operational criteria, mission success, technical performance, cost, schedule, quality, risk, failure rate, maintainability, and supportability. Through definition and prioritization of all design solution objectives, the optimal solution is achieved that best satisfies the requirements set under consideration.

Often, devices perform their functions at varying performance levels in differing environments. For instance, the system delay for a computer system gathering surveillance data from various sources and formulating a graphical representation of all existing air vehicles in a given space and presenting it to the controller on a display is vastly different at various locations and at various times during the day. Stakeholders would only state minimum National Airspace System (NAS) requirements for presentation of data to them from the source. The engineers involved in Synthesis must decide how they will meet those stated requirements in the various environments. A tailored system for each location might be provided, thus lowering the overall cost of upfront procurement, since computer systems with less processing power may be used in small airport areas. However, training and support regarding multiple systems must also be addressed in terms of added cost for multiple versions of the system. In this example, the Synthesis engineers must evaluate the operating environment of the solution to determine the performance objectives, upfront procurement cost, and the lifecycle costs of supporting the resulting system. These items represent three distinct objectives to be satisfied in selecting a design that will fulfill the stakeholder needs.

Another facet to consider is that a single system design may not necessarily satisfy all of the requirements associated with the functional area under consideration. Multiple systems may be required to satisfy the entire requirements set.

Ideally, alternative solutions should satisfy all requirements, but it is useful to include solutions that challenge the requirements and lead to a better system concept. Various options are to be considered eventually in light of the objectives for the resulting system(s). Such alternatives include relaxing requirements of marginal utility that are costly to implement or extending requirements when added capability can be purchased cheaply while accruing operational benefits.

4.5.3.1.1 Performance Objectives

The performance objectives, although highly dependent on potential system solutions, must be clear, as they serve to define the main purpose of the system. The engineering team must not only define all terms that will measure how the system will perform, but it must also state the actual desired performance levels. The team must review and analyze the accuracy, capacity, response time, throughput, and other similar requirements against feasible design possibilities. The threshold performance levels are clearly documented for the design under consideration. Most, if not all, of the performance requirements are contained in the program requirements provided under Requirements Management. However, the stated performance objectives that are to be achieved by any potential system or systems are clearly documented at the outset of Synthesis so that the tradeoff between these and other objectives may follow.

4.5.3.1.2 Reliability Objectives

The engineering team must define the reliability objectives in terms of the likelihood or probability that the resulting system will operate at its objective performance level for a defined period of time under normal operating conditions. In clearly defining these objectives, engineers must translate the environmental and operational data, such as the data in the OSED. Allocation of the Reliability, Maintainability, and Availability (RMA) requirements in the program requirements is conducted in concert with the requirements process and Specialty Engineering in order to allocate the various reliability maintainability objectives to the various design alternative functional areas.

4.5.3.1.3 Compatibility Objectives

The engineering team must define the objectives to enable the system to work or interface with both existing systems and those under agency development. Interface objectives are stated in terms of interfaces (including physical and functional descriptions (see Interface Management, Section 4.7, subsection 4.7.1.2)), but also in terms of the working environment imposed by the existing systems or system elements with which the potential design alternative must interact. The objectives must address both backward compatibility with legacy systems and forward compatibility with known evolving technologies, protocols, and standards.

4.5.3.1.4 Flexibility Objectives

The engineering team must define the objectives to **enable alternative design approaches to adjust to a changing environment**. For example, the ability to process more flight data to adapt to a growth in air traffic must be clearly defined and documented. This is particularly important when it is known that the existing environment will evolve. The design alternative must evolve with the environment to adapt to the new environment. Projections for changes are documented along with the stated objectives for flexibility of the design alternative.

4.5.3.1.5 Extensibility Objectives

Extensibility differs from flexibility, which means the ability to adapt to and accommodate growth needs. **Extensibility is the ability of the design alternative to serve new or multiple uses.** An example of extensibility is a multipurpose display that provides graphical display of flight plan data, surveillance data, or both simultaneously without need for modification.

4.5.3.1.6 Cost Objectives

A limited budget is a never-ending facet of the Synthesis process. Thus, it is essential to define clearly the cost objectives at the outset for any potential design alternative. Try not to overemphasize cost of the item over all other objectives. The old adage, “You get what you pay for,” is all too often true. Consequently, cost objectives are best stated within a range for the design alternatives. Cost objectives must include all facets of the potential design alternatives’ lifecycle. Restricting objectives merely to the initial cost of a design solution may not fairly consider other design alternatives that have higher initial cost, but whose overall lifecycle costs are lower due to quality, reliability, and supportability characteristics. Therefore, the cost objectives shall be defined for all stages of the intended lifecycle.

4.5.3.1.7 Schedule Objectives

What a design alternative will do, how well it will perform the function(s), and where it will perform become irrelevant if the design alternative is not delivered to the user when needed. A design alternative delivered too early is as potentially damaging to the effort as one delivered too late. Therefore, the schedule objectives for all facets of the design alternatives’ lifecycle must be defined clearly and comprehensively. The schedule objectives for test, operational introduction, full operational capability, service life, and so on are all documented.

4.5.3.1.8 Identify Objectives Tradeoffs and Define Objectives Hierarchy

Rarely, if ever, do projects have unlimited time and financial resources. Tradeoffs and compromises are common during Synthesis in order to achieve the design objectives with an acceptable level of requirements compliance. It is essential to define the design objectives and rank their relative importance.

The prioritized set of objectives—defined during the brainstorming, brainwriting, and dynamic confrontation meetings—is to be well established and documented before design solutions are considered.

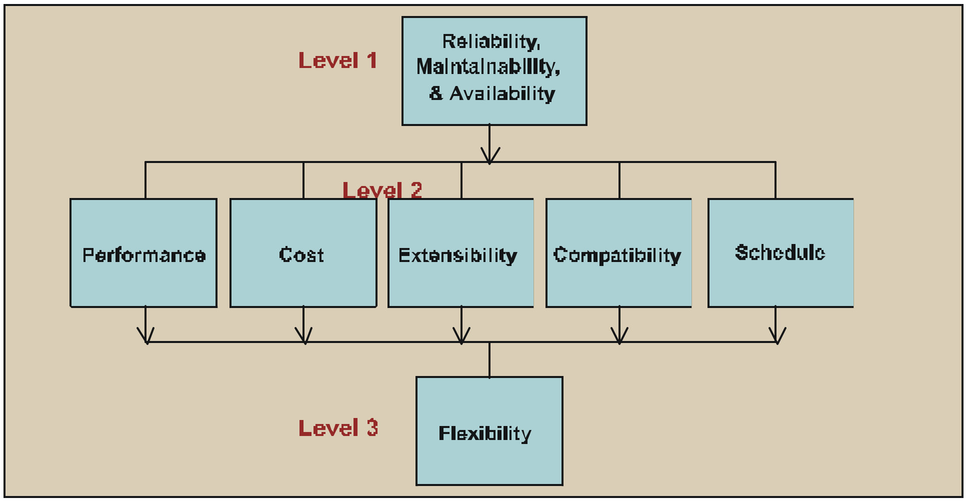

Figure 4.5-4. Example Three-Level Objectives Hierarchy

Objectives from the above categories and additional categories to be considered under the program/project are first documented as a list. The list is expanded to include more categories as determined necessary in concert with program management, Specialty Engineering, and stakeholders. The importance of each objective relative to the others is then determined for all objectives. Once all the relative priorities are established, priority levels are defined based on the findings. This task, although not simple, is necessary because the results are invaluable later when design alternative tradeoff analysis is performed.

Assume that each of the categories of objectives just described has one objective; there are then a total of seven resulting objectives. For this example, examine a project that eliminates a reliability deficiency in an existing fielded system. In this particular example, RMA is therefore considered more important than all other alternatives. Also, since the product introduced is only an interim solution to fulfill a shortfall, system flexibility is considered less important than all other factors. If all remaining objectives are considered to be of equal importance, there are three priority levels (Figure 4.5-4)

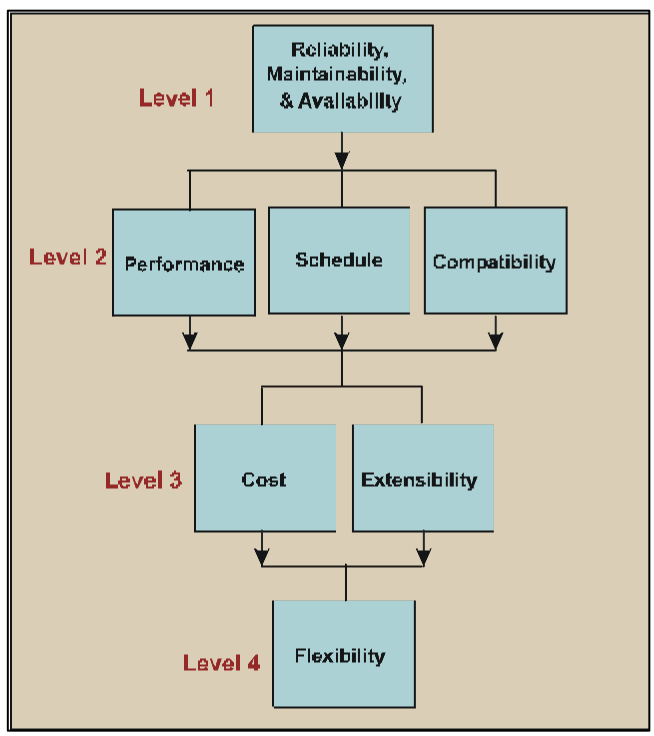

Figure 4.5-5. Example Four-Level Hierarchy

Establishing the objectives hierarchy is seldom this simple. The items in level two of the figure are rarely seen as equal in importance. This level may be further broken down into groups, with each group containing objectives of equal importance and with one group being considered to be more important than the other. This leaves four levels of priorities instead of three, and the hierarchy is established, with relative objective priorities and priority-level definition (Figure 4.5-5).

4.5.3.2 Identify Potential Design Alternatives—Step 2

During this step, grouping of needed functions into common functional areas is complete, and the functional architecture is established. The design team must now begin partitioning desired requirements into design elements. In reviewing various designs regarding whether or not they will perform the desired functions, the team maps each requirement, grouped functionally in the functional architecture, to a component of the system under review. Some components will satisfy one requirement, whereas others may satisfy more (Figure 4.5-6).

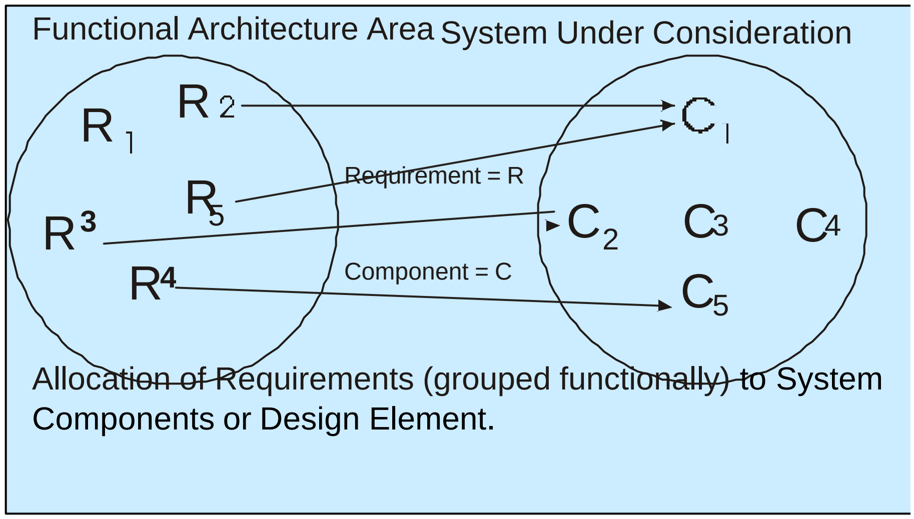

Figure 4.5-6. Functional Partitioning to System Components

This Synthesis process step boils down to generating alternative design solutions for the functional elements identified during Functional Analysis (Section 4.4) that perform the needed functions and adhere to the requirements for that functional area. The alternative solutions should be composed of one or a combination of more than one of the following: hardware, software, material, data, facility, people, and techniques.

There are a variety of tasks conducted to identify an array of design alternatives. Various subteams may perform the tasks sequentially or concurrently. If the Synthesis team is small, it is best for all members to consider identifying alternatives sequentially. If the team is large enough and good communications exist among all members, the team should explore concurrently identifying solutions by the various means described in the subsequent subsections. Both approaches require that the entire group conduct prior planning. Concurrent exploration of alternatives requires close coordination throughout identification of alternatives until all possibilities are identified; whereupon, the subteams will once again combine to complete this Synthesis step. Figure 4.5-3 (above) illustrates the tasks feeding the Synthesis step that identifies the various design alternatives.

4.5.3.2.1 Identify Technology Requirements

This assessment addresses not only potential incorporation of existing technology into design solutions, but also looks at the risks and limits imposed by and on that technology. Each alternative being considered is analyzed against the changing technologies available in the marketplace. Available technologies are studied for use in the design under consideration, potential improvements to design performance, improvement to maintainability of the resulting system, cost-effectiveness, and maturity.

The need for a new technology that makes possible a performance or functional improvement previously not possible must be carefully weighed against the risk imposed by that technology. The potential benefits of inserting the technology must outweigh the potential risks to cost, schedule, and performance.

To continue consideration of the potential technology insertion, the impacts to the end user must be considered through human factors analysis. The tasks, roles, and jobs assigned to humans are analyzed and assessed to discover whether the end users of the resulting system have the required knowledge, skills, and abilities. If the knowledge, skills, and abilities do not exist, then the cost and schedule risks of achieving them with the new technology are weighed against the benefits derived from the technology. Training and personnel pipelines are fully evaluated to ensure that they meet requirements.

4.5.3.2.2 Identify System Specialty Engineering Attributes

The design team must work in concert with specialty engineers to identify the characteristics of each potential alternative necessary to fulfill interdisciplinary needs.

The design team and specialty engineers work together to:

- Analyze each alternative
- Identify potential hazards to system hardware/software components
- Identify the humans involved in the system as users or support personnel
- Identify characteristics of the proposed operational environment

The analysis must demonstrate that the design under consideration results in safe system operations. The analysis includes all aspects of the design, development, manufacture, test, operation, and support of the potential design.

The design team works with human engineering to analyze each alternative for human factors suitability. Each alternative is analyzed regarding the human user system interface. (See Specialty Engineering (Section 4.8).)

4.5.3.2.2.1 System Safety Engineering

System hazards are identified and assessed for the design alternative. The hardware, software, operational, and ambient environments, as well as procedures and human elements of the design alternative, are analyzed. Historical or test data is applied to estimate the risk (severity and likelihood) of each identified hazard. Controls are then designed in accordance with the safety order of precedence described in Specialty Engineering (Section 4.8) subsection 4.8.1. All hazards and their associated controls are prioritized according to their risk criticality rating. The analysis results are used to direct further design efforts to characterize controls, safety features, redundancy, and system degradation elements of the system.

4.5.3.2.3 Identify Off-the-Shelf Opportunities

Each design alternative is analyzed to determine if an off-the-shelf item exists that will fulfill the allocated requirements. Off-the-shelf solutions can include non-developmental hardware or software.

Once off-the-shelf solutions are identified, each must be assessed to ensure that a variety of factors are considered in determining suitability. The number of systems available off the shelf must be gauged against the number that users need. The quantity required must include not only those needed initially by the user community, but also those needed to serve as replacements over the anticipated service life of the system.

Another facet of the suitability assessment process is consideration of the environment in which the prospective off-the-shelf item must eventually operate. The proposed item must be able to adapt to the existing support structure to be suitable. If the item requires new equipment and/or training for support during its lifecycle, the benefits of the item must outweigh its cost and schedule impacts.

Finally, the manufacturer(s) of the off-the-shelf item must be assessed. Attributes such as product maturity, upward/downward compatibility, manufacturer track record, financial stability, and quality practices must be factored into the commercial product selection process. If the products or manufacturers fall short in any of the reviewed categories, they must be considered a risk. Refer to Appendix F of the FAA COTS Risk Mitigation Guide (at http://www.faa.gov/aua/resources/cots/intro.htm) for a more detailed listing of COTS nontechnical selection factors.

4.5.3.2.4 Identify Make-or-Buy Alternatives

A cost analysis is performed for the design alternative(s) and used to support a make-or-buy decision. This analysis must address whether it is more cost-effective to produce the design element or use an established supplier.

When cost, schedule, and risk are considered, the best choice is to design and develop (a "make" decision) a singular system that satisfies all functional area requirements. The team will proceed with this approach as a viable design alternative.

4.5.3.3 Define Solution Set—Step 3

Input from prior processes and previous Synthesis steps identify not only potential alternatives, but also design constraints for potential solutions. This input is used to help determine if existing or newly developed items can accomplish the function under consideration.

Synthesis strives to identify viable design alternatives, refine those alternatives to fulfill the program requirements, and finally select the most balanced and beneficial design to introduce into the field. To accomplish this goal, all possible alternatives are first identified. These are reduced to reflect only those alternatives considered viable or worth pursuing.

4.5.3.3.1 Populate the Solution Set

The design team identifies all possible design solutions that may serve to satisfy all or part of the program requirements. After exploring and then exhausting these possibilities, team members, as a group and individually, evaluate the design solution set. If only one possible design alternative has been identified, then the job is not complete. No matter how large or difficult the program requirements and their associated functional area are, there will always be at least one possible design alternative: *do nothing*—that is, continue the status quo and not present new and/or innovative design solutions. Given the fact that a great effort went into previous SE processes (such as Requirements Management (Section 4.3) and Functional Analysis (Section 4.4)), it is unlikely that entrance into Synthesis would have occurred if all requirements in the functional area, with its associated program requirements, were satisfactorily met. Clearly, it is possible to identify an insufficient number of alternatives. The task is to develop additional alternatives that present better options.

The following methods can be used to develop new alternatives.

- **Change the characteristics of existing alternatives.** First, list all existing alternatives and then itemize the main characteristics of each. Generate a table with the rows representing the list of alternatives and the columns representing the main characteristics of all alternatives. In all likelihood, each of the potential alternatives will possess characteristics that are both similar and distinct from those of the other alternatives. Identify the positive characteristics and then list the missing characteristics needed by a design alternative and not represented by any potential solution. Finally, add more alternatives to the list, since the characteristics within the previously listed alternatives are varied. This addition enhances the new alternatives with needed positive characteristics and eliminates as many negative characteristics as possible.

- **Return to the objectives.** Focus on the most important objectives one at a time and list alternatives that will meet each of those top-level objectives. Then, work down the objectives hierarchy, developing more alternatives or refining existing alternatives that satisfy those additional objectives.
- **Finally, examine all the objectives and requirements set.** List alternatives that will maximize the number of objectives and requirements that can be met with the alternative.

If there still seems to be a lack of viable alternatives, step through the various methods, introducing more creativity and ingenuity each time through. Eventually, a solution set will reach a stable point, and identification of design alternatives is complete.

Having identified a significant number of design alternatives, one must now evaluate all alternatives. First, determine that a number of sound viable design alternatives exists that can satisfy all or most of the program requirements. It is possible to continue the Synthesis process with too many design alternatives because the remaining steps will detail and document each alternative to a great degree. Therefore, continuing with too many alternatives can waste valuable time and resources. One can argue that proceeding with one alternative is not sufficient. Likewise, one might also argue that proceeding with 10 alternatives that must be thoroughly defined and documented is unnecessarily excessive; so, reducing the alternatives set to a manageable size or number of alternatives (based on the scope of the stakeholder need) is a must.

4.5.3.3.2 Reduce Solution Set to Manageable Number of Alternatives

When viable design solutions are identified, one must not compromise **requirements considered absolutely necessary to satisfy the operational needs. These requirements—which a system must meet or be deemed unnecessary or unacceptable—are to be considered "*threshold requirements.*"** A potential design solution must satisfy threshold requirements for further consideration as a design alternative. Threshold requirement compromise or tradeoff is not an option. A design alternative not meeting a threshold requirement that cannot be modified easily to meet the requirement(s) is eliminated and not considered further.

The objectives hierarchy is used next. If the remaining alternatives set contains potential solutions that do not meet the top-level objectives—and they cannot be easily or affordably modified to do so—then they are eliminated from the set of potential alternatives. As with requirements, some objectives are not subject to compromise, and alternatives not meeting the high-priority objectives, as defined earlier, should no longer be considered.

If potential solutions are only able to satisfy a portion of the functional area requirements or objectives, consider various options to develop a set of viable design solutions. One or more of the solutions that nearly satisfy the objectives and/or requirements could be modified to achieve satisfactory results. The following options may be used to modify either the problem (functional area under consideration with its associated requirements) or the alternative design solutions.

- **Request Trade Study.** A detailed analysis, such as that conducted under Trade Studies (Section 4.6) is requested to determine if one or more of the options can be modified to fulfill the desired requirements and/or objectives. Under the Trade Studies process, incorporation of new technologies and a variety of other means are investigated. If the results of the study render viable design alternatives, then Synthesis proceeds to the next step, requirements allocation. However, if no alternative can meet all of the requirements in the functional area under consideration, the requirements and/or the functional areas are analyzed.

- **Initiate Requirements Feedback.** When the program requirements for the functional area under consideration cannot be satisfied through viable design alternatives, feedback to Requirements Management (Section 4.3) is initiated. If program requirements are only partially met by all potential designs, Synthesis and Requirements Management concurrently analyze the ability of the alternative solution to meet the requirements set. Consideration is given to modifying requirements to lower and achievable levels. Full compliance is deferred until technological or other advances allow for full compliance with the original requirements. Requirements that cannot achieve even partial compliance in the various designs are addressed through the design loop.
- **Initiate Design Feedback**. Due to discovery of design issues, the Functional Analysis (Section 4.4) is reexamined, and the initial decomposition or performance allocations are reassessed. Design issues include identifying a promising physical solution or open-system opportunities that have different functional characteristics than those foreseen by the initial functional architecture requirements. Issues also include the inability of all design alternatives to fulfill the same functional architecture; this may be addressed by repartitioning the functional area. The functional area is subdivided so that allocation of those requirements to be satisfied by the alternative designs can be made down to perspective system elements. The remaining functional areas whose associated requirements will not be satisfied remain with the Functional Analysis (Section 4.4) process. The associated requirements are documented as unsatisfied in the Requirements Management (Section 4.3) process. The functional area(s) with the associated unsatisfied requirements are partitioned out of Synthesis back to Functional Analysis for future Synthesis loop identification of potential solution(s).

All remaining alternative solutions are reviewed and analyzed in concert with Specialty Engineering, risk management, lifecycle engineering, and integrated program planning to determine adequacy and suitability of each remaining alternative. The alternatives are pared down to preferred design solutions.

4.5.3.4 Allocate to System Elements—Step 4

The previous Synthesis steps have resulted in a promising set of conceptual designs for systems satisfying the program requirements for the functional area under consideration. Each design concept must now be developed in more detail so that requirements and design constraints are assigned to the top-level elements of that system design.

4.5.3.4.1 Allocate Requirements to System Elements

In prior steps, the functional area and associated requirements were adjusted in concert with Functional Analysis and Requirements Management, respectively. As this Synthesis step is entered, the program requirements to be satisfied by the design solution(s) are established, and this step furthers the design process by allocating the requirements to system elements.

These elements are the highest level distinct elements of the system in the areas of hardware, software, and humans in the system. Each system element must perform at least one function within the functional area to be considered separately and distinctly in the traceability of requirements.

The design engineers allocate program requirements to the selected system elements. They document all program requirements that the system must satisfy and formally begin tracking the requirements through the various design and acquisition phases of the system. Documentation includes information regarding the hardware, software, or other system components to which each requirement is allocated.

4.5.3.4.2 Allocate Design Constraints to System Elements

Design constraints that apply directly to system elements are identified. These constraints do not apply to the functions performed, but rather to the elements: hardware, software, or people. Design constraints differ from constraint requirements in that they recognize existing limitations to design of a system, its interfacing systems, and its operational and physical environment. Such design constraints will include power, weight, data throughput rates, memory, and other resources. These constraints represent the system's inability to achieve a capability or level of performance due to such issues as insufficient technology and lack of available facility space for the system.

Design constraints are especially important in analyzing the design of potential replacements for existing systems. This is of particular interest to design engineers when major elements of the original system may be retained. Once allocated, the design constraints will clearly define which system elements remain, are added, or modified.

The technology constraints identified during the prior technology assessment are allocated to the system elements. Those constraints identified during review of Specialty Engineering attributes are also allocated to ensure that inappropriate design characteristics are not introduced into the selected system. Finally, environmental constraints are allocated down to the system element level. Environmental constraints can be introduced by climatic conditions under which the total system will operate, by the facilities in which the system will be housed, or more globally by environmental hazards and constraints (such as Environmental Protection Agency regulations) imposed in the region(s) where the systems will be used.

4.5.3.5 Define Design and Performance Characteristics—Step 5

With the system concepts now defined, identify and document the design and performance characteristics of each alternative. Characterization of the system(s) is all-inclusive and addresses all facets of the system under design, including the associated human-engineering elements and lifecycle considerations or needs.

During this phase, there is substantial benefit to practice concurrent engineering. The entire functionality of the system(s) under design is considered. When the design and performance characteristics are defined, the entire lifecycle of the potential system must be considered—from inception to disposal—in an integrated process. This requires involvement of all Specialty Engineering disciplines (Section 4.8) in the Synthesis process. Thus, sound engineering decisions are made based on strong consideration of all phases and aspects of the system under design consideration.

4.5.3.5.1 Assess Failure Modes and Effects

Failure modes and the effects of failure are assessed for the design alternative. The hardware, software, and human elements of the design alternative are analyzed, and historical or test data is applied to estimate the probability of successful performance of each alternative. Use a failure modes and effects analysis (FMEA) to identify the strengths and weaknesses of the design solution. (See Reliability, Maintainability, and Availability Engineering (subsection 4.8.2) of Specialty Engineering, Section 4.8.) For critical failures, a criticality analysis is conducted to prioritize each alternative by its criticality rating. The analysis results are used to direct further design efforts to characterize redundancy and graceful system degradation elements of the system.

4.5.3.5.2 Assess Testability Needs

The design team analyzes the testability of the design in relation to the operational or maintenance needs. The team determines the need for a built-in test, Remote Maintenance Monitoring, and/or a fault-isolation test for each potential design alternative. Test mechanisms are considered in the design and incorporated as necessary for elements that are normally maintained by the users or field support engineers. Diagnostic operations to support lower level maintenance actions are likewise incorporated into the design solution.

4.5.3.5.3 Assess Standardization Opportunities

The alternative is assessed for possible use of standardized end items that are technologically and economically feasible. Use of design elements that implement commercial and international standards is strongly considered.

4.5.3.5.4 Assess Lifecycle Factors

The design of each alternative is assessed to determine the degree to which quality factors (producibility, ease of distribution, usability, supportability, trainability, and disposability) have been included in the solution. Additionally, associated lifecycle process needs, requirements, and constraints are identified and defined for each design under consideration. (See Lifecycle Engineering (Section 4.13).)

4.5.3.6 Define Physical Architecture—Step 6

A physical architecture defines and describes the way in which the various functional architecture elements can be assembled to form physical entities. The physical entities must represent a viable design alternative and must provide one or more services that address user needs as translated by the program requirements. The physical architecture may involve such physical entities as runways and various forms of equipment; such nonphysical entities as software; or a combination of the two.

The physical architecture identifies the physical subsystems, and architecture flows between subsystems that will implement the functions and provide the needed services/capabilities. The physical architecture further identifies the system inputs and outputs.

In constructing a physical architecture, use the following definitions.

- **Physical Entities.** The classes of physical entities that will be used are:
  - *Subsystems*. Subsystems are the primary structural components of the physical architecture. They perform functions that "belong" together and whose interfaces require interoperability and compatibility. It is a system in and of itself (reference the system definition) contained within a higher level system. The functionality of a subsystem contributes to the overall functionality of the higher level system. The scope of a subsystem's functionality is less than the scope of functionality contained in the higher level system.
  - *Users*. These are people who interact with the architecture implementation. They could be either those who use the system (such as the flying public or pilots in the NAS) or operators who use features of the system (such as air traffic controllers in the NAS). Each interface to a user involves human interaction with the system.
  - *External Systems*. These are organizations and agencies (such as Department of Defense or National Weather Service external to the NAS) and/or their systems that will likely interact/interface with the system under design.
  - *Environment*. This is the physical world (e.g., pavement, air, obstacles).
- **Physical Interfaces.** These are mechanical, electrical, data, and other interfaces between system elements or subsystems. Physical interfaces also include all interfaces between the system and its outside world.

4.5.3.6.1 Decompose Into Physical Entities

The architecture can be viewed at several levels of detail. The architecture defines collections of subsystems while defining their interfaces. Consideration is given to a variety of engineering and programmatic disciplines along with stakeholder contributions, and all are incorporated into the physical architecture.

4.5.3.6.2 Define Physical Interfaces

Identify and define the physical interfaces among products, subsystems, humans, lifecycle processes, and external interfaces to higher level systems or interacting systems. Physical interfaces that impact design include communication, data, support, test, control, display, connectivity, or resource replenishment characteristics of the interaction among subsystems, the products, humans, or other interfacing systems or a higher level system. (See Interface Management (Section 4.7).)

4.5.3.7 Analyze and Refine Design Alternatives—Step 7

As a particular design alternative is refined, it is analyzed to determine how it satisfies the allocated functional and performance requirements, interface requirements, and design constraints and how it adds to the overall effectiveness of the system or a higher level system. During analysis, specialty engineers work with design engineers to ensure that requirements such as reliability, availability, maintainability, supportability, safety, human factors, security, electromagnetic compatibility, and spectrum management are incorporated into the design. Additionally, lifecycle process requirements are identified and defined for each alternative system product solution and aggregate of solutions.

4.5.3.7.1 Assess Design Capacity To Evolve

The design alternative is analyzed regarding its capacity to evolve or be reengineered, accommodate new technologies, enhance performance, increase functionality, or incorporate other cost-effective or competitive improvements once the system is in production or in the field. Limitations that may preclude the system's ability to evolve should be identified and the approach analyzed and refined to resolve any limitations. The supportability of an evolving system may require the support process to evolve along with the product. This consideration may significantly affect support funding and training requirements.

4.5.3.7.2 Develop Models and Prototypes

Models and/or prototypes are developed to assist in:

- Identifying and reducing risks associated with integrating available and emerging technologies
- Verifying that the design solution (consisting of hardware, software, material, humans, facilities, techniques, data, and/or service) meets allocated functional and performance requirements, interface requirements, workload limitations, and constraints
- Verifying that the design solution satisfies functional architecture and program requirements

The models, data files, and supporting documentation are maintained, and each version of a model or data file that impacts requirements, designs, or decisions is saved in the integrated database. Models may be digital, partial, or complete and may be hardware, software, or a combination of both; or they may include human models or human-in-the-loop simulations or mockups for usability testing and workload measurement. (See Trade Studies (Section 4.6).)

4.5.3.8 Assess Requirements Compliance—Step 8

Compliance with the program requirements for the functional area is reviewed and analyzed. For each alternative, the solution level of compliance to all requirements is documented. If none of the alternatives achieves full compliance, and all fail to meet the same requirements, the design loop is initiated. If some, but not all, of the alternatives fail to fully meet all of the requirements, and compliance varies among approaches, the requirements feedback loop is initiated for each design. This is not to be confused with Verification (Section 4.12).

4.5.3.8.1 Perform Design Loop

The design loop involves revisiting the functional architecture to verify that the physical architecture developed is consistent with the functional and performance requirements. It is a mapping between the functional and physical architectures. During Synthesis, reevaluation of the Functional Analysis may be caused by discovery of design issues that require reexamination of the initial decomposition, performance allocation, or even the higher level requirements. These issues might include identification of a promising physical solution or open-system opportunities that have different functional characteristics than those foreseen by the initial functional architecture requirements.

4.5.3.8.2 Perform Requirements Feedback Loop

The system design is audited to determine compliance with the program requirements set. Audits are performed at various levels, from the top-level physical architecture down through each hierarchy level to the lowest-level system element or configuration item. Compliance with program requirements is assessed through both informal and formal reviews. The audit results are then fed back to earlier Synthesis steps as needed, resulting in another Synthesis loop. The audit results may call for program requirement changes at varying levels, or they may lead to design changes to ensure compliance.

4.5.3.9 Select “Best Value” Alternative—Step 9

The “best value” alternative must be the one that offers the most balanced design. The “best value” alternative is selected using all prior analysis conducted in Synthesis or in conjunction with Requirements Management (Section 4.3), Functional Analysis (Section 4.4), Trade Studies (Section 4.6), Specialty Engineering (Section 4.8), and Risk Management (Section 4.10). Upon being selected, the design is detailed and finalized. The designation and description of interfaces (internal and external) among design elements are finalized. The design is baselined and placed under formal configuration management processes.

4.5.4 Process Outputs

It bears repeating that Synthesis is an iterative process, concurrent with Functional Analysis (Section 4.4) and Requirements Management (Section 4.3). The engineering team must use good judgment in aligning the degree of detail of the Synthesis outputs with the position of the project in the AMS cycle.

Prior to selection of the ”best value” alternative, Synthesis outputs are completed concisely and at a very high level for *all* possible solutions. As the functional analysis and program requirements become more specific, there will be fewer and fewer alternative solutions that answer the need. As the process narrows toward the “best value” alternative, the top choices will have detailed, documented outputs from the Synthesis team. Once the Joint Resources Council chooses the preferred solution, the Synthesis team will complete the definition of the design process down to the very finest detail.

Therefore, the following Synthesis outputs occur throughout the iterative process, but they vary in scope and detail based on the project’s position within the AMS cycle.

4.5.4.1 Physical Architecture

For all the alternative solutions, the system elements are identified along with their arrangement and the interactions between them. A description of the salient features of the overall solution is developed as well as descriptions for the system elements and their relationships establishing a potential system architecture baseline. The descriptions are diagrams, schematics, concept drawings, tabular data, and narrative reports.

The design architecture is established at a level that documents the design solution and interfaces. It includes the requirements traceability and allocation matrices, which capture allocation of functional and performance requirements among the system elements. Design architecture definitions should be stored in the integrated database along with tradeoff analysis results, design rationale, and key decisions to provide traceability of requirements up and down the architecture. Verification of the design architecture should be done to demonstrate that the architecture satisfies both the validated program requirements and the verified functional architecture. This information is further compiled into a Requirements Compliance Matrix.

4.5.4.2 Description of Alternatives

4.5.4.2.1 Concept Description Sheets

A separate description for each of the alternatives developed and refined during Synthesis is documented. For the selected or preferred design, more detail is provided to enable other SE processes to best use the information. The description sheets include a complete description of the system, the system operational use, and characteristics.

4.5.4.2.2 Architecture Block Diagrams (ABD)

The ABD documents the hierarchical relationship of all system elements. The ABD includes hardware and software elements and their hierarchy, documentation and data, facilities, test equipment, and support.

An external ABD is also to be developed to depict the external elements that affect the selected system. Like the system ABD, the external ABD should include all hardware, software, facilities, personnel, data, and services having a significant effect on the selected system.

4.5.4.2.3 Schematic Block Diagrams (SBD)

The SBD illustrates the physical partitioning and interfaces for each viable candidate hardware and software design solution. SBDs should not be developed for every conceivable design—only for those that are worthy of detailed evaluation (based on position within AMS cycle).

4.5.4.2.4 Interface Drawings

Drawings are developed for all system physical element interactions as well as for all interactions to external physical elements. The drawings provide a mental picture of interfaces and are the basis by which interface requirements and control documents are developed later under Interface Management (Section 4.7).

4.5.4.3 Product Definition

The drawings, schematics, software documentation, manual procedures, and so on are developed as necessary to document the selected design elements in a product definition.

4.5.4.3.1 Configuration Item Descriptions

Each of the system elements is identified during the Synthesis process. This includes all hardware configuration items (HWCI) and computer software configuration items (CSCI). Each HWCI and CSCI is documented and described at the time of its summary or preliminary identification. Once the “best value” alternative is selected, detailed documentation for each

HWCI and CSCI of the selected system is developed, thus establishing a configuration baseline for the system. (See Configuration Management (Section 4.11).)

4.5.4.3.2 Specification Inputs

During Synthesis, compliance with the program requirements (Requirements Verification Compliance Document (RVCD)) was assessed. This analysis sometimes results in recommendations for modification or elimination of requirements. Any proposed modifications or deletions are documented and forwarded to Requirements Management (Section 4.3).

4.5.4.3.3 Requirements Compliance Matrix

All requirements have been mapped to the system elements. As the mapping occurred during Synthesis, a matrix was developed containing all requirements, the subsystem or element to which they were assigned, and the level of adherence to the requirements achieved by the system component. The matrix is designed for each level of the physical architecture, and it lists all performance, functional, and constraint requirements to reflect each level of the architecture. Compliance levels are determined using system/cost-effectiveness analysis, simulations, demonstrations, inspection, and/or testing.

4.5.4.3.4 Refined Work Breakdown Structure (WBS)

The selected design's physical architecture is used to refine the WBS by translating the decomposition into a WBS format. The refined WBS provides enhanced work planning, cost/schedule tracking, and control by extending the existing WBS to account for the system elements identified during Synthesis.

4.5.4.4 Constraints

Constraints are formed before the program enters the Synthesis process, and yet more may be identified during the process. Synthesis looks at many different aspects of the system design, including cost, scheduling, feasibility, requirements, function, and others. As various solutions to the Mission Need Statement are considered and refined, constraints become apparent.

Constraints are clearly seen when performing step 4, "Allocate to System Elements" (subsection 4.5.3.4), of the Synthesis process. The constraints identified may cause iteration through the design feedback loop or the requirements feedback loop. An evolutionary development is initiated, if necessary, for any design element for which a lesser technology solution was selected over a higher risk technology, and for which the capacity to evolve was designed into the element and interfacing elements. (See Trade Studies (Section 4.6)).

4.5.4.4.1 Design Constraints

Step 5, Define Physical Architecture (subsection 4.5.3.5) identifies and documents constraints specific to the Synthesis process. These design constraints do not apply to the functionality of the system, rather they are in the area of hardware, software, or people. Because these design constraints are so important in analyzing replacement of existing systems, they are documented and sent on for further study in the Lifecycle Engineering process (Section 4.13), aiding in identifying the timing of future replacement schedules. Additionally, these design constraints become another output of the Synthesis process, as requests for a Trade Study (Section 4.6) evaluation are sent out.

4.5.4.5 Planning Criteria

Planning criteria describing planned activities for the Synthesis process are output to the Integrated Technical Planning process (Section 4.2).

4.5.4.6 Tools/Analysis Requirements

Tools/Analysis Requirements for performing Synthesis throughout the remainder of the program's lifecycle need to be provided to the Integrity of Analysis process (Section 4.9).

4.5.4.7 Concerns and Issues

Appendix D contains guidance on Concerns and Issues as a product of Synthesis and how to best convey that information to the Risk Management team (Section 4.10).

4.5.5 Metrics

Performance of the Synthesis process itself shall be measured on a regular basis and recorded in the metrics library monthly. The following metrics, at a minimum, will be used to evaluate performance:

1. Trade Study Satisfaction Assessment (see Trade Studies (Section 4.6))
2. For approved engineering problem reports:
    a. Quantity, by type of problem report
    b. Cycle time from disposition to incorporation of change into released engineering documents, by type of report
3. Technical Performance Measurements: objective versus achieved values
4. Number of approved engineering changes: by product, type, and stage
5. Documents/drawings submitted for engineering release:
    a. Unacceptable submittals
    b. Total submittals
6. Number of technical action items identified during reviews and audits
7. Design efficiency metrics, such as weight, required power, and envelope dimensions (volume)
8. Cost and schedule variance for completion of Synthesis steps
9. System requirements not met
10. Number or percent of system requirements verified by system analyses
11. Number of items yet to be determined within the system architecture or design
12. Number of interface issues not resolved
13. Percent of identified system elements that have been defined

4.5.6 Tools

4.5.6.1 Schematic Block Diagrams

Along with the definition of design alternatives, it is important to establish the relationships between alternatives at each level of design activity. One can use SBDs to accomplish this.

A simplified SBD shows the components that may comprise an element and the data that may flow between them. An expanded version is usually developed that displays the detailed functions performed within each component and their interrelationships. For complex systems, this may then be developed into a logic diagram for auditing the schematics produced. This audit is a critical SE function. Interface information should also be embedded into the SBDs, as appropriate. The interface data will form the basis for the interface specifications to be developed at multiple levels of the system hierarchy. An N-squared (N^2) diagram (see Functional Analysis (Section 4.4) for examples) is extremely useful for developing and auditing interfaces at all levels.

If software is an element of the design, it must be determined whether a given function will be accomplished in hardware or software. Computer Software Elements (CSE) should be defined during this step of the process and embedded within the SBDs. Experience shows that it is helpful to first define the top-level HWCI and/or CSCI in which a given software function will reside before defining which candidate CSEs will accomplish the function. Additionally, as part of subsection 4.5.3.6 (Define Physical Architecture) of the Synthesis process, it is recommended that a given function be tracked to determine whether it has been allocated to a software alternative or a hardware alternative. Determining the appropriate level of the system hierarchy for defining CSEs is largely project dependent.

The products of this step of the SE process are a set of viable system alternatives responsive to the design goals and a series of SBDs depicting how the alternatives interrelate.

4.5.6.2 Computer-Aided Design

Modern computing hardware and software are used to convert the initial idea for a system into a detailed engineering design. The evolution involves creating geometric system models that are later manipulated, analyzed, and refined.

4.5.7 References

1. Blanchard, Benjamin S. *Systems Engineering Management*. Second Edition. New York, NY: John Wiley & Sons, Inc., 1998.
2. Defense Systems Management College. *Systems Engineering Fundamentals*. Fort Belvoir, VA: Defense Systems Management College Press, 1999.
3. *European ITS Framework Architecture, The KAREN Project—Physical Architecture*, Version 2. European Communities, Keystone Architecture Required for European Networks (KAREN), August 2000. (http://www.frame-online.net/Karen_doc/d36a1fin.pdf)
4. *Guidelines for Approval of the Provision and Use of Air Traffic Services Supported by Data Communications*. DO-264. Prepared by SC-189. Washington, DC: Radio Technical Commission for Aeronautics, Inc. (RTCA), December 14, 2000.
5. *IEEE Std-1220-1998, IEEE Standard for Application and Management of the Systems Engineering Process.* Los Alamitos, CA: Institute of Electrical and Electronics Engineers, 1998.
6. *International Council on Systems Engineering (INCOSE) Systems Engineering Handbook*. Version 2.0. Seattle, WA: INCOSE Central Office, 2000.
7. Martin, James N. *Systems Engineering Guidebook: A Process for Developing Systems and Products*. Boca Raton, FL: CRC Press LLC, 1997.
8. *National ITS Architecture Documentation, Physical Architecture*. Lockheed Martin Federal Systems, Odetics Intelligent Transportation Systems Division. Washington, DC: ITS America, September 1998.
9. *Processes for Engineering a System*. EIA-632. EIA Standard. Arlington, VA: Electronic Industries Alliance, January 7, 1999.
10. Sage, Andrew P., and Rouse, William B. *Handbook of Systems Engineering and Management.* New York, NY: John Wiley & Sons, Inc., 1999.

11. *Systems Engineering Capability Model*. EIA/IS-731.1. EIA Interim Standard. Arlington, VA: Electronic Industries Alliance, December 1998.

12. *Systems Engineering Guidebook for ITS*. Version 1.1. California Department of Transportation, Division of Research and Innovation, February 2005, page 197.